

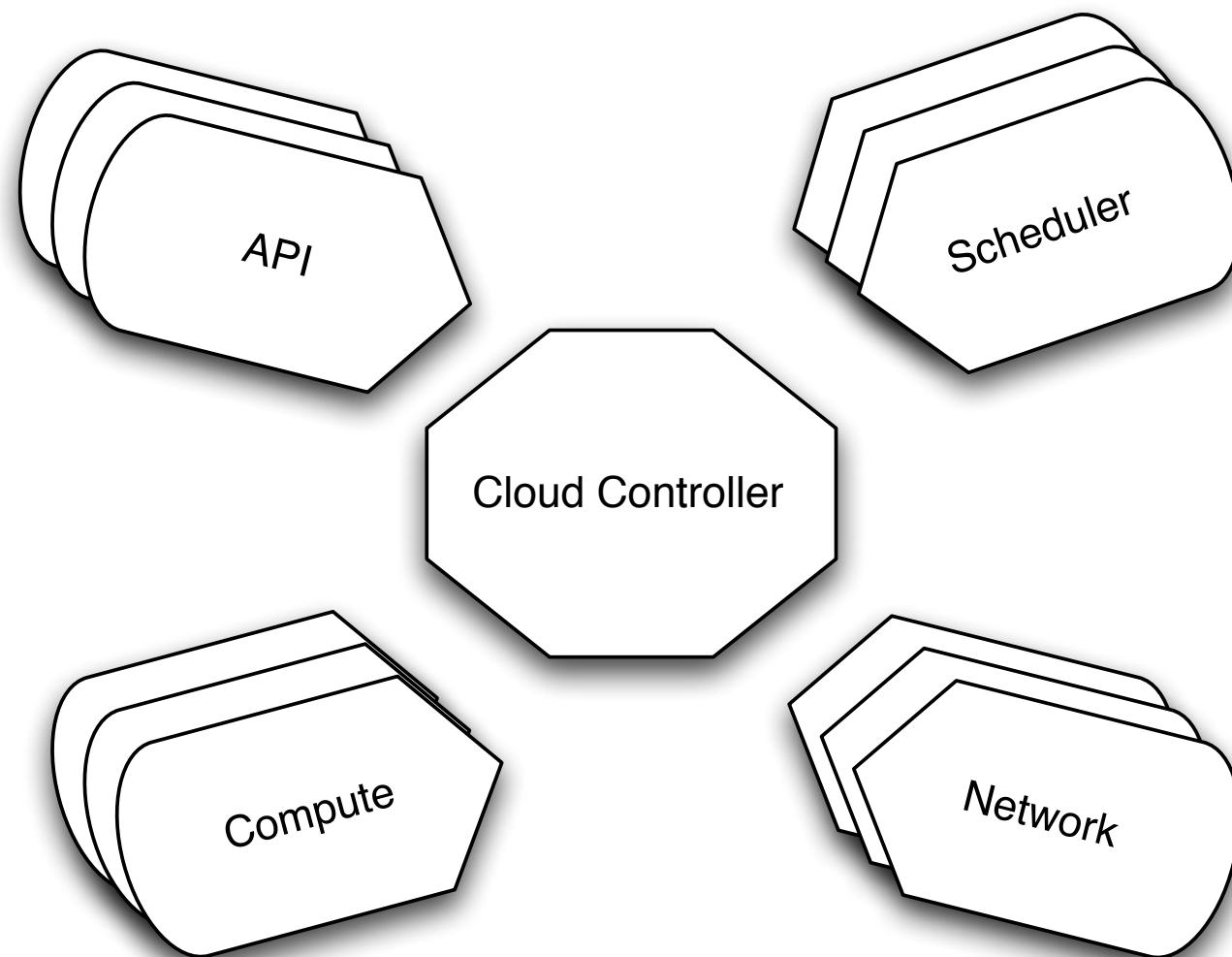
# OpenStack Compute

Codenamed: Nova



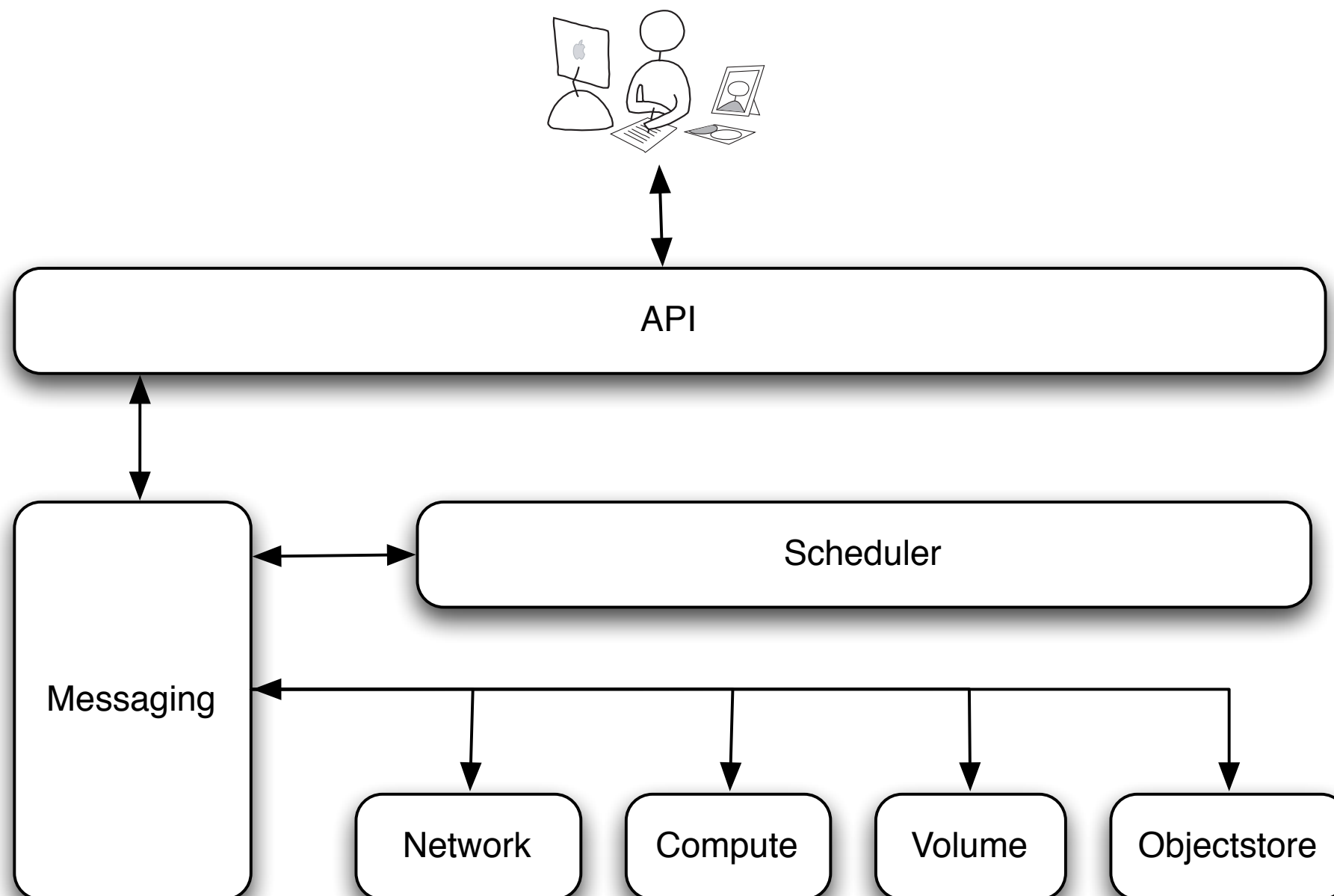
# Nova Components

# Nova Components



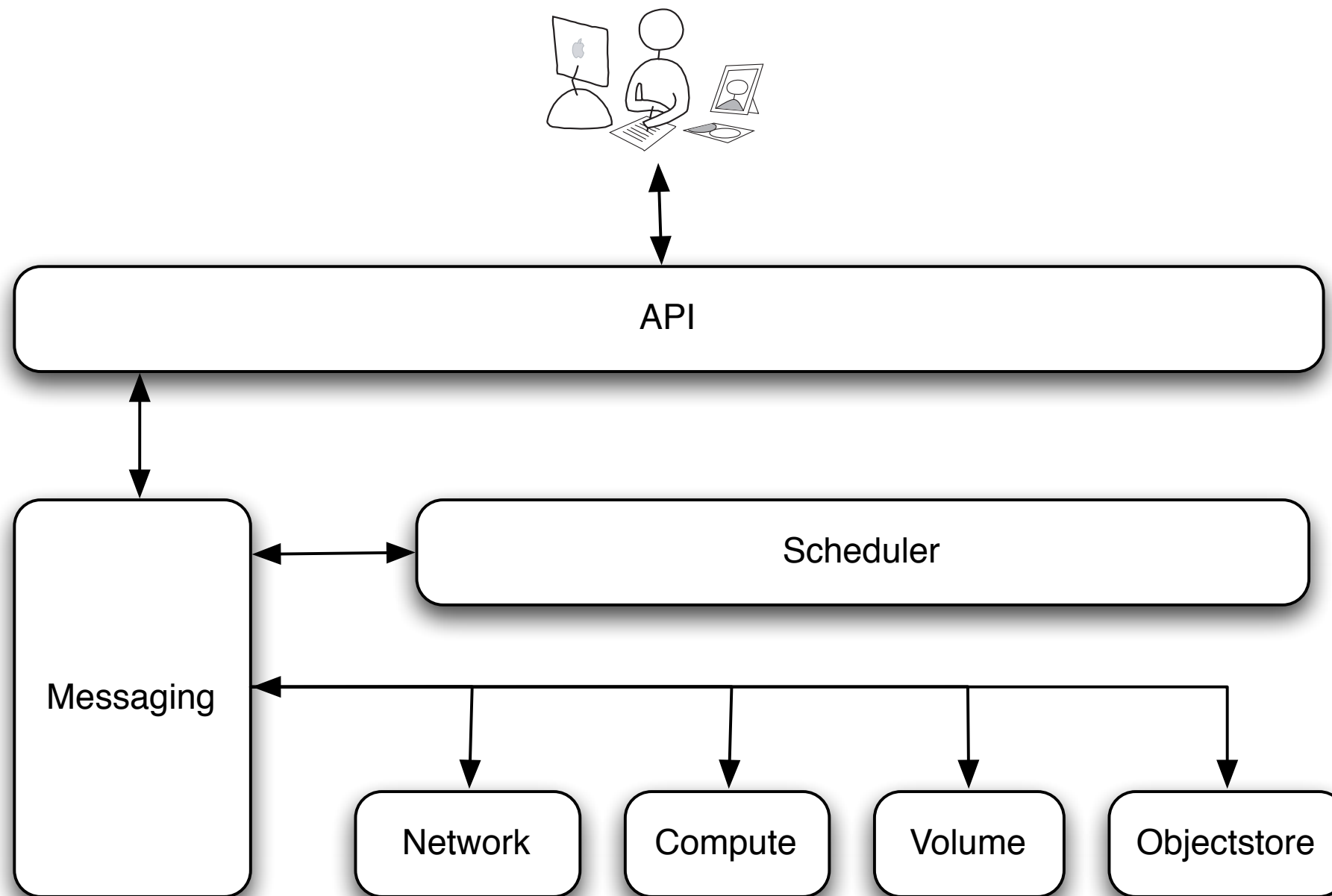
# Nova Components

## Request Flow

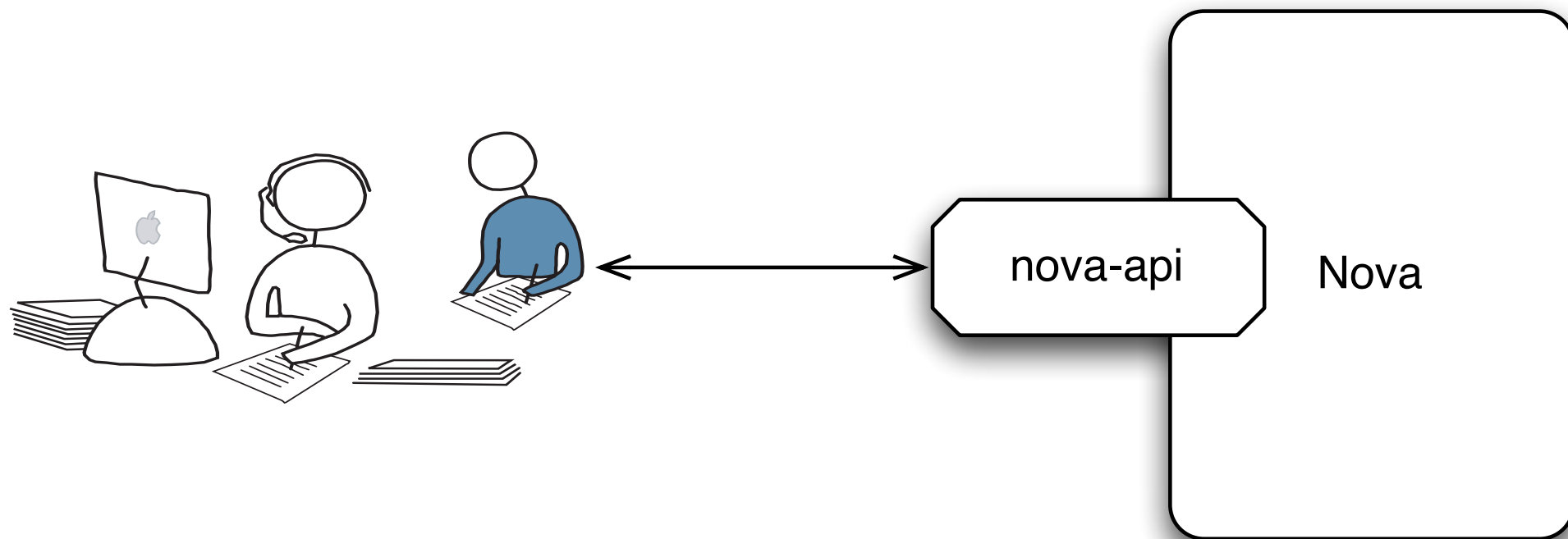


# Nova API Node

# API Node

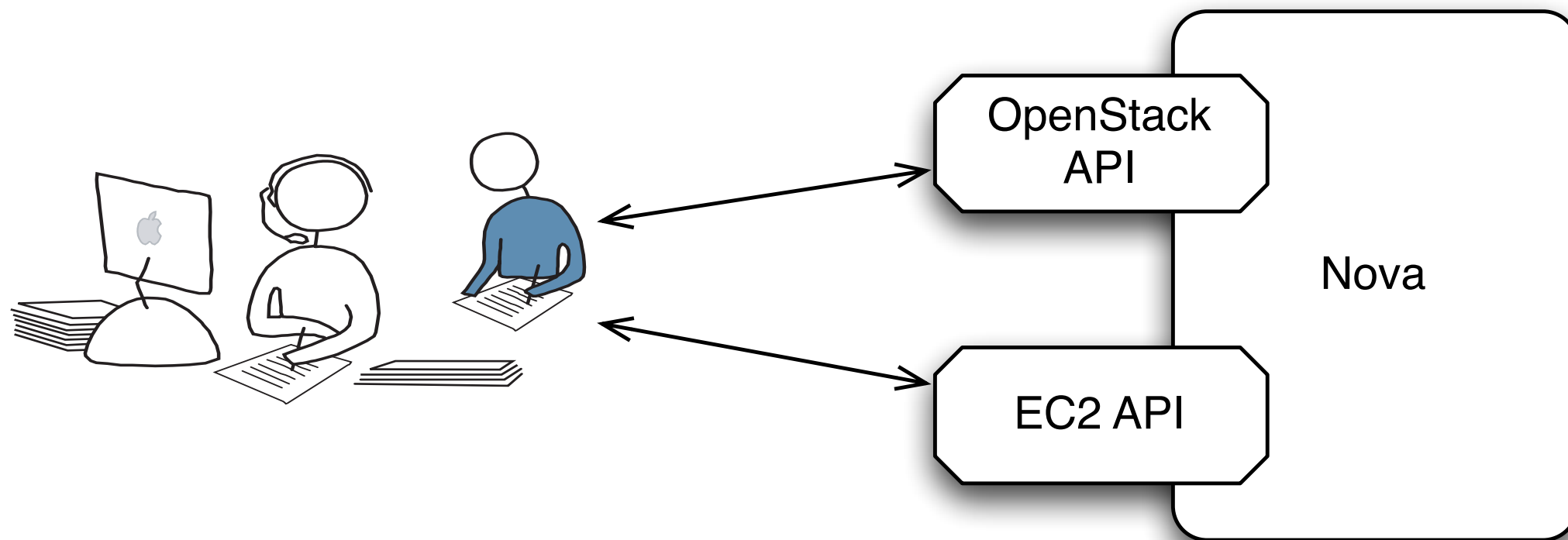


# API Node



- An API Node is a machine running the nova-api service
- Serves as the primary gateway to Nova

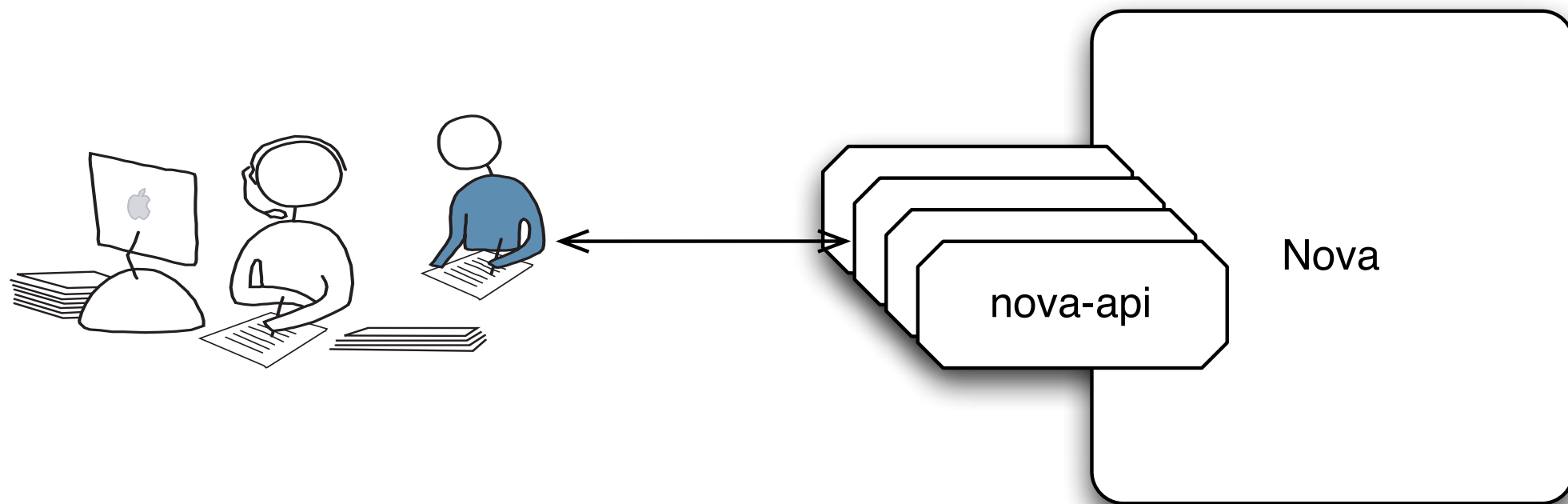
# API Node



- Supports the OpenStack API and the EC2 API
- EC2 API is support for backwards compatibility
- The OpenStack API is preferred when possible

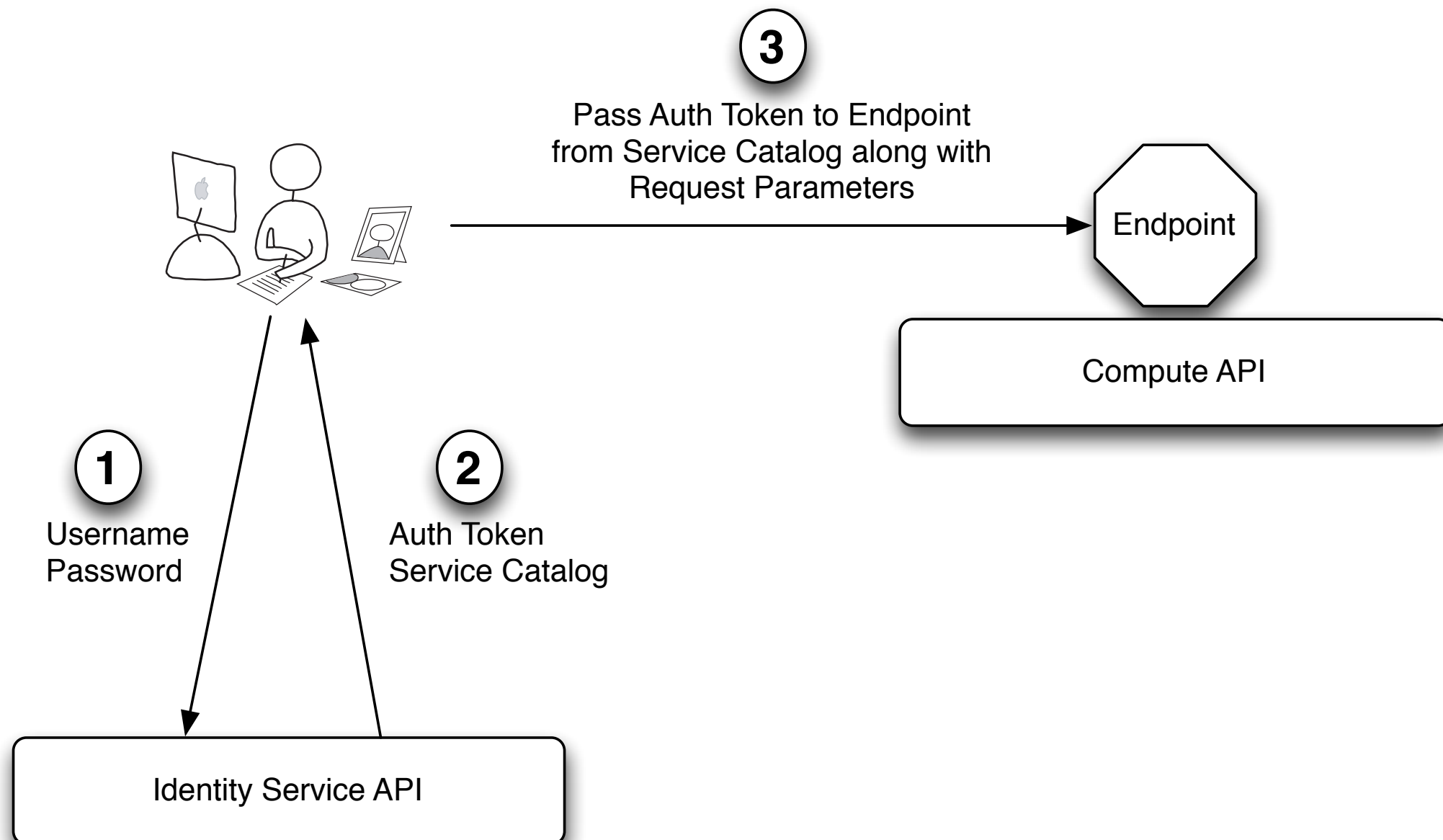


# API Node



- A Nova installation has one or more API Nodes

# API Flow



# On Top of the API

Horizon

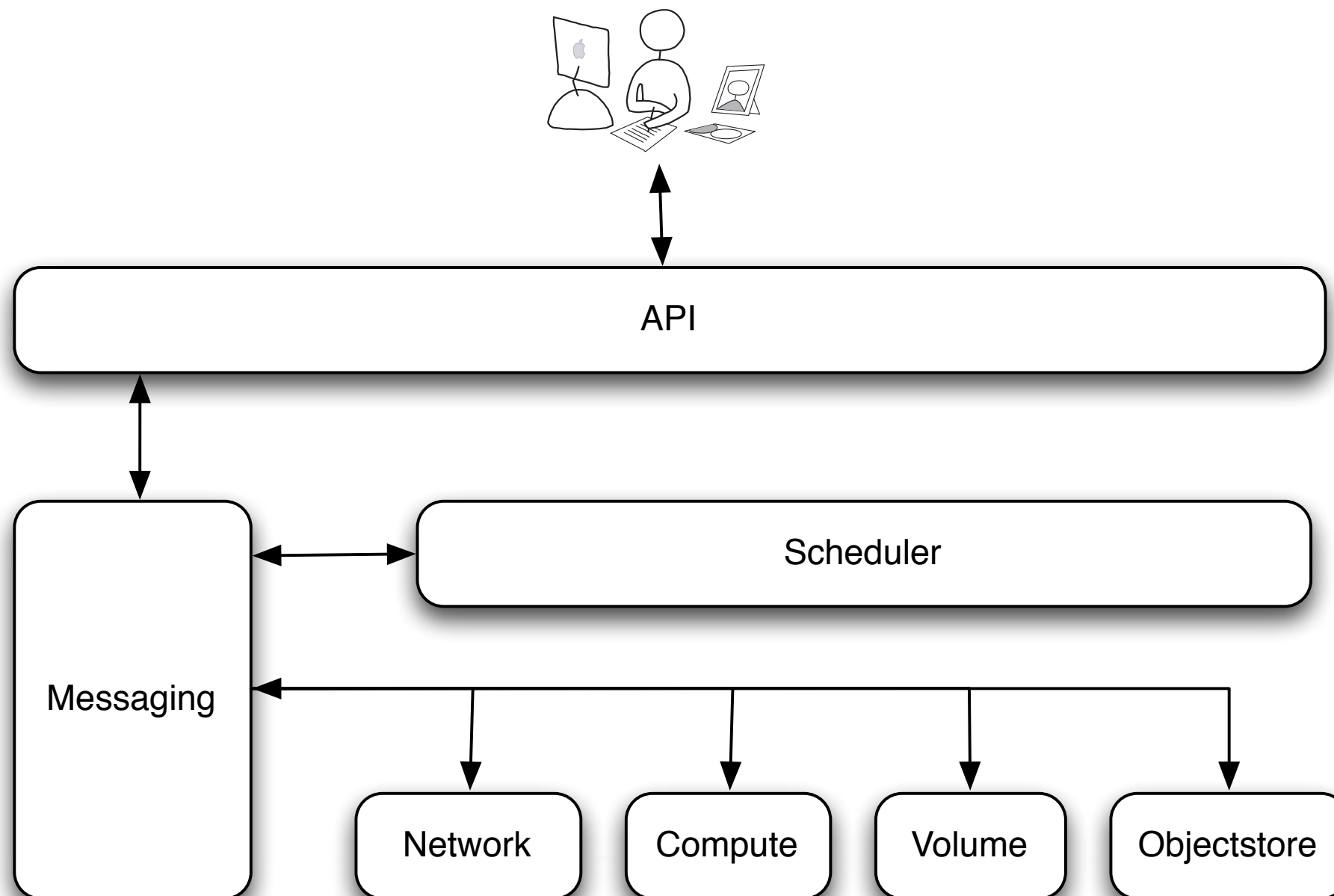
nova-manage

nova

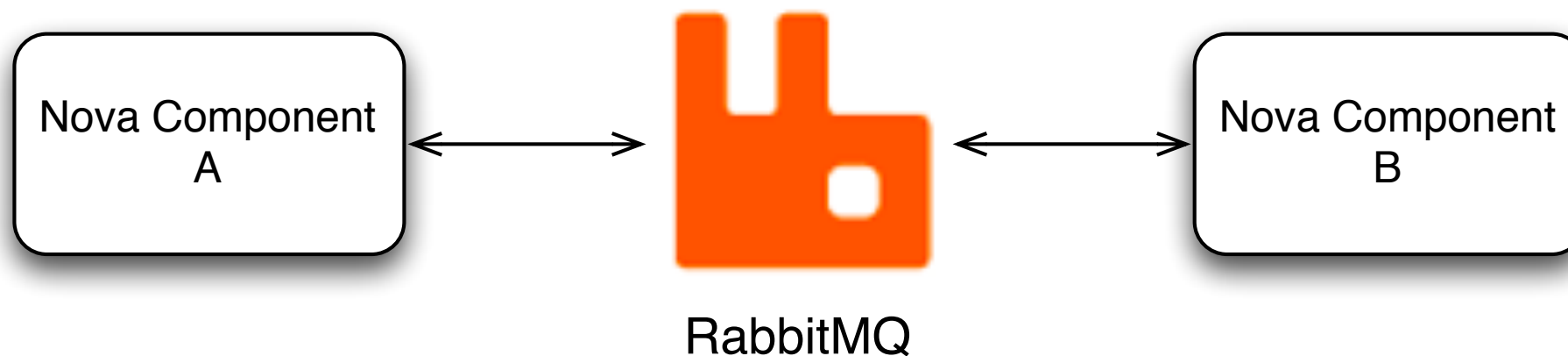
Compute API

# Nova Messaging

# Messaging



# Messaging

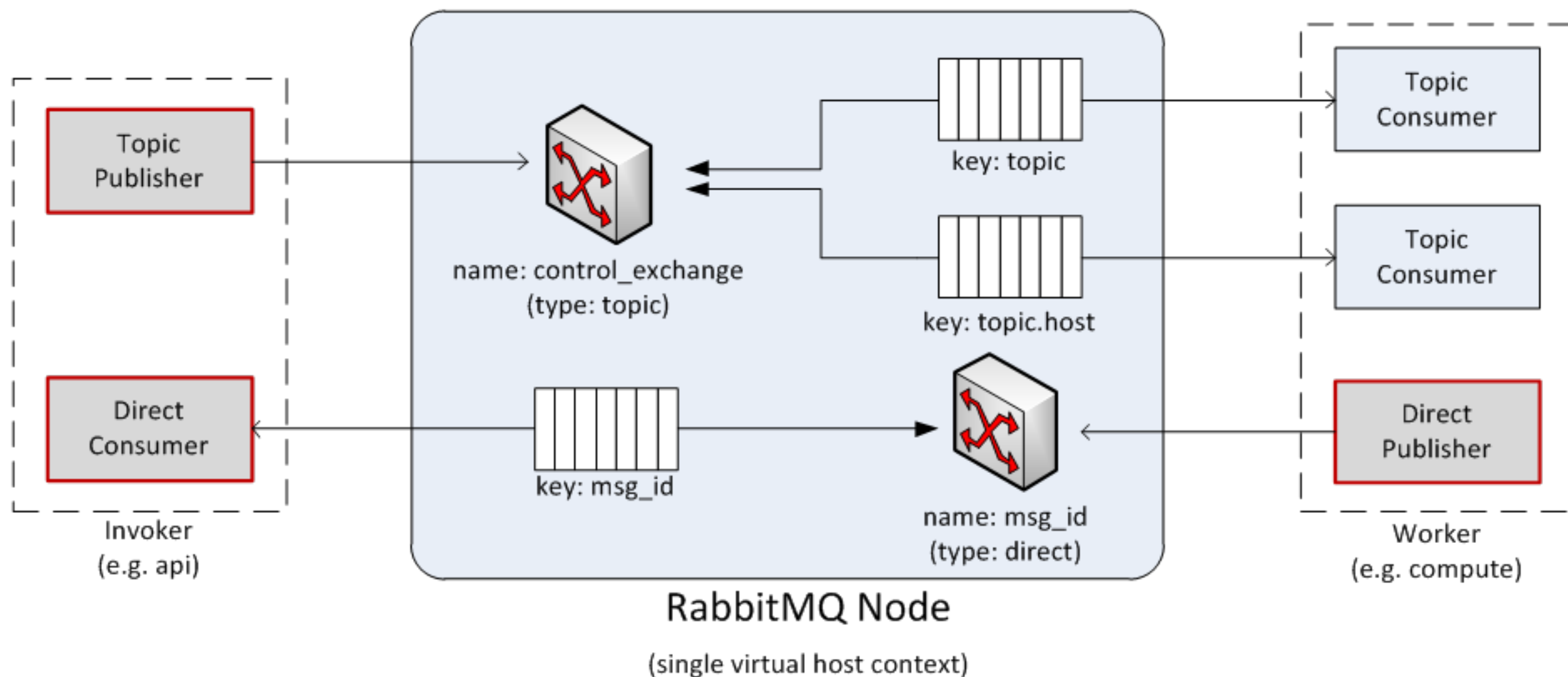


- OpenStack uses the RabbitMQ messaging platform
- RabbitMQ sits between any two Nova components
- Allows components to be loosely coupled

# Messaging



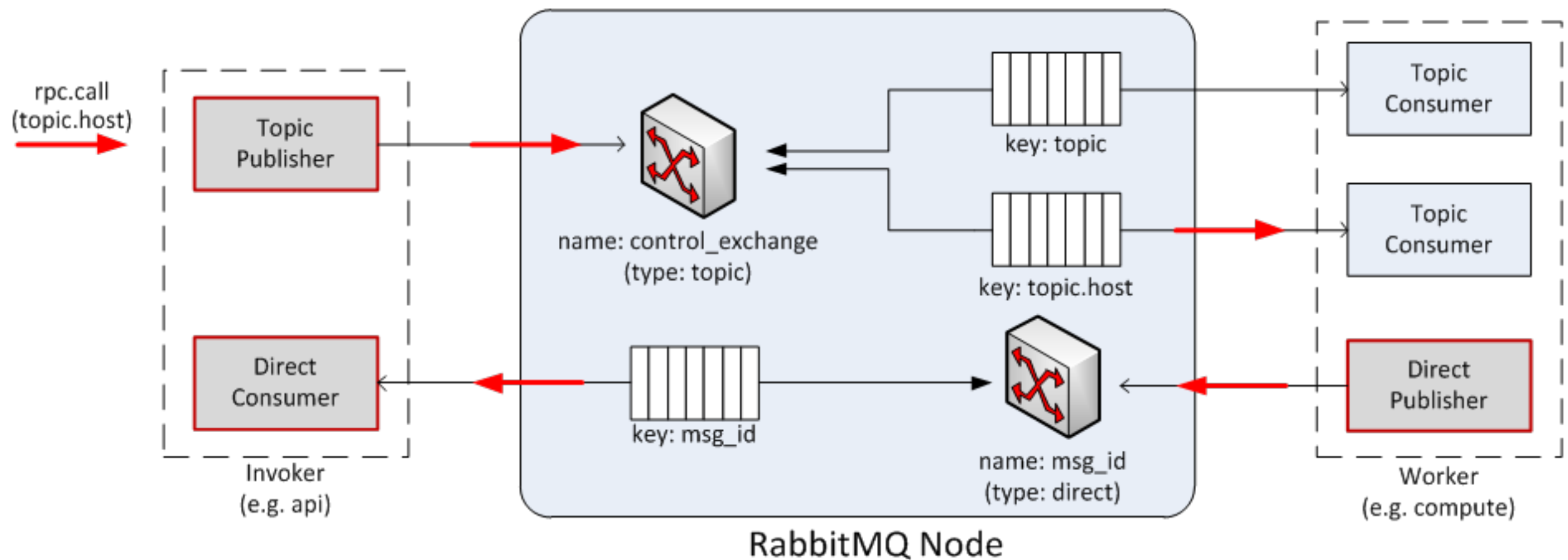
# Messaging





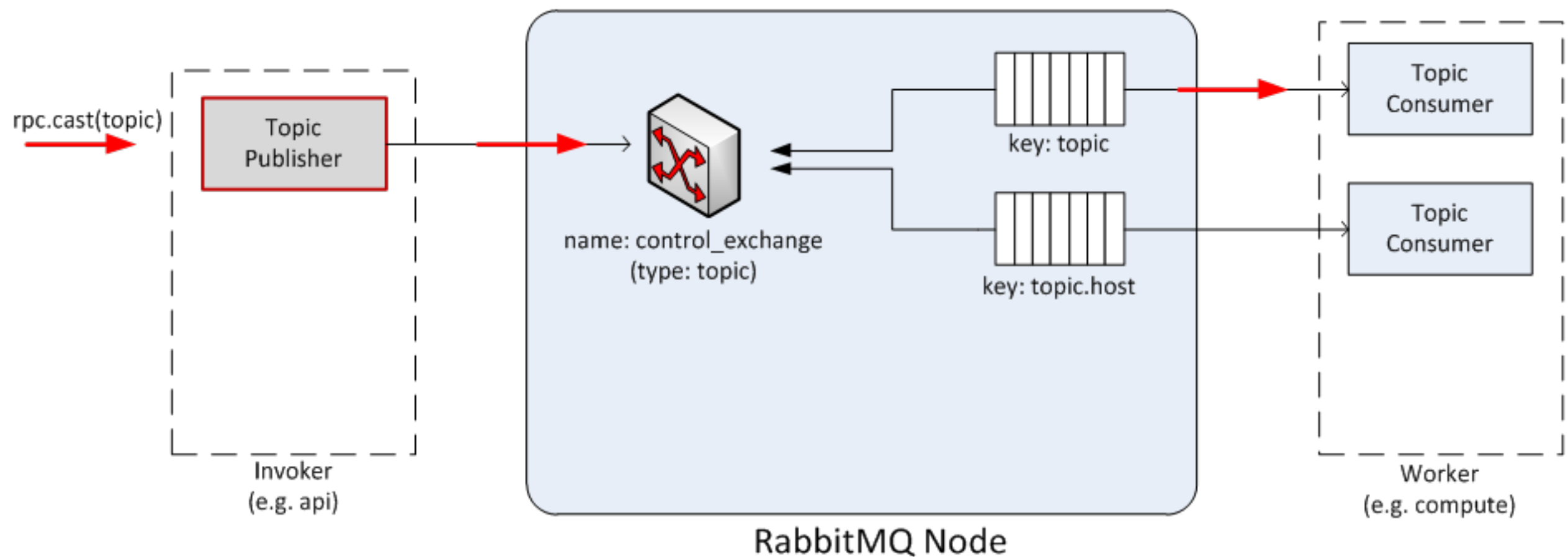
# Messaging

## rpc.call



# Messaging

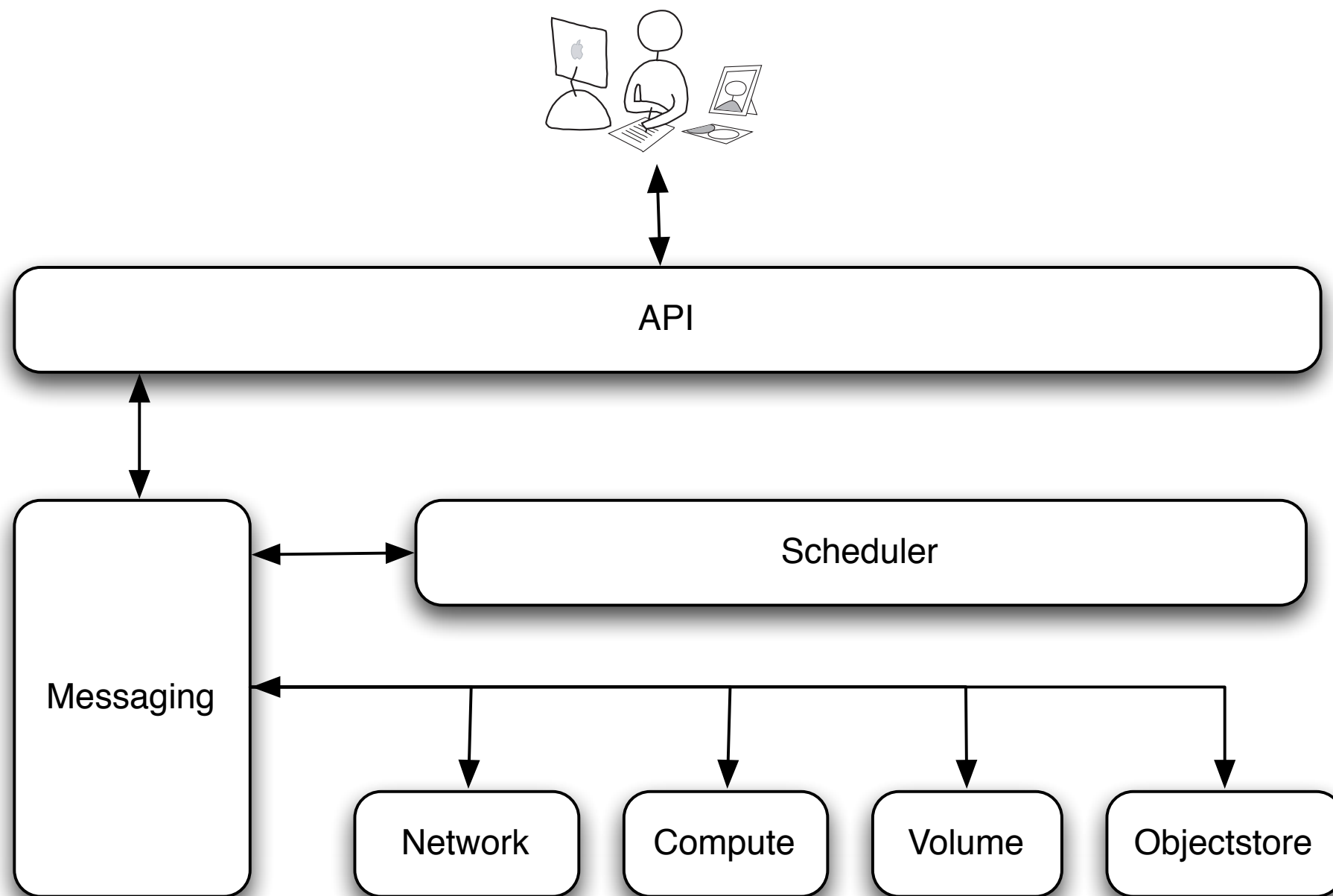
## rpc.cast



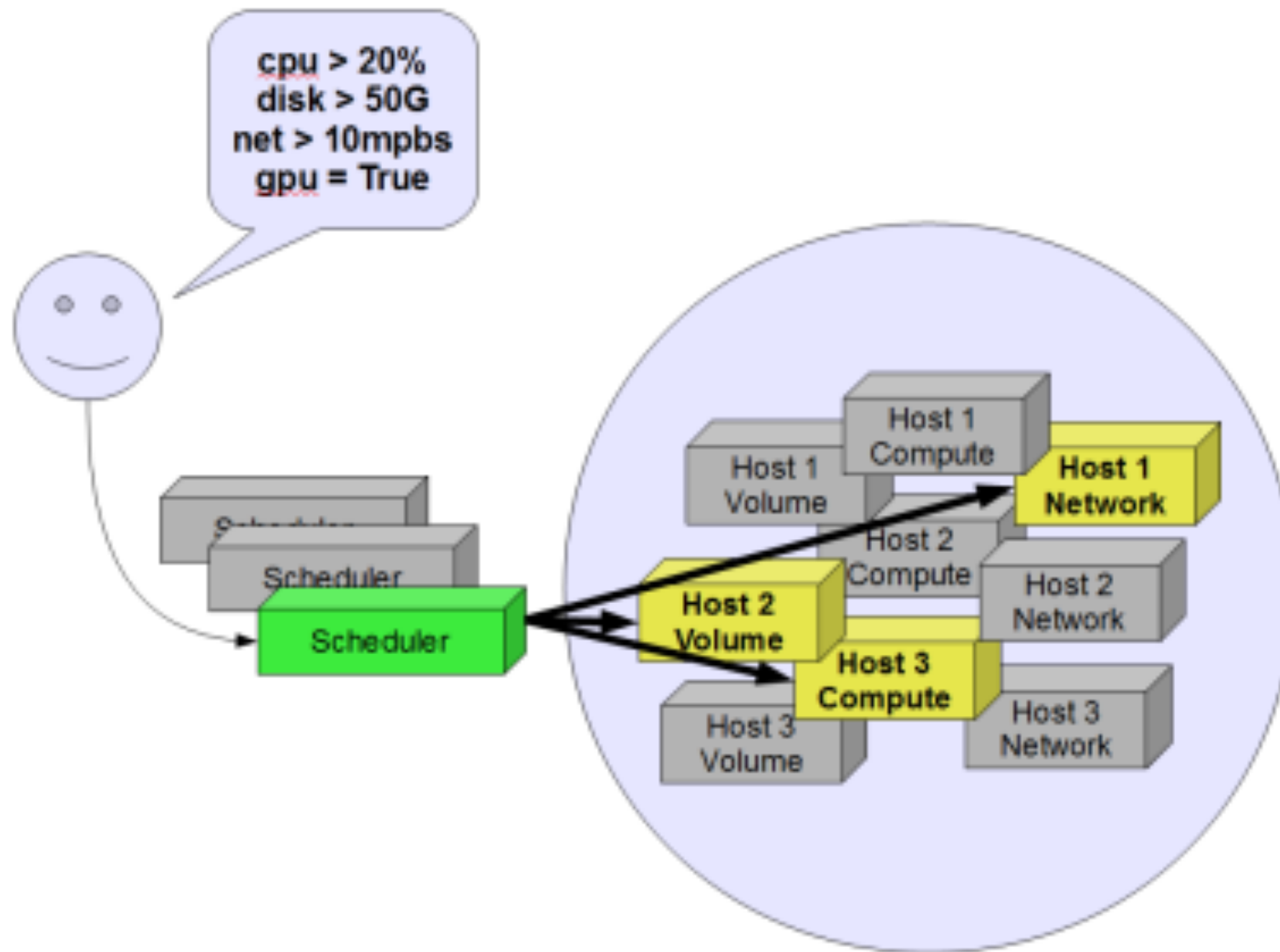
# Nova Scheduler

# Nova Components

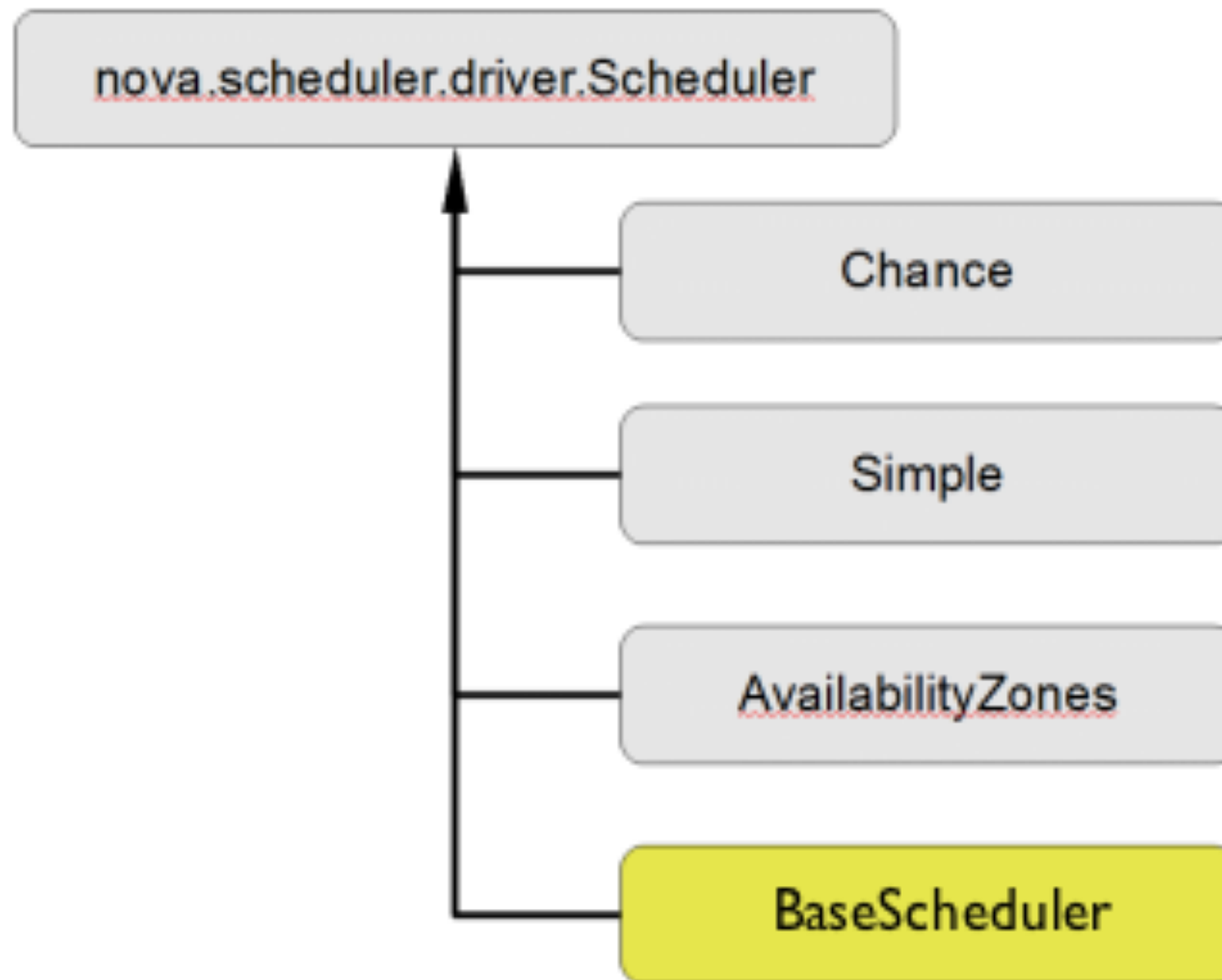
## Request Flow



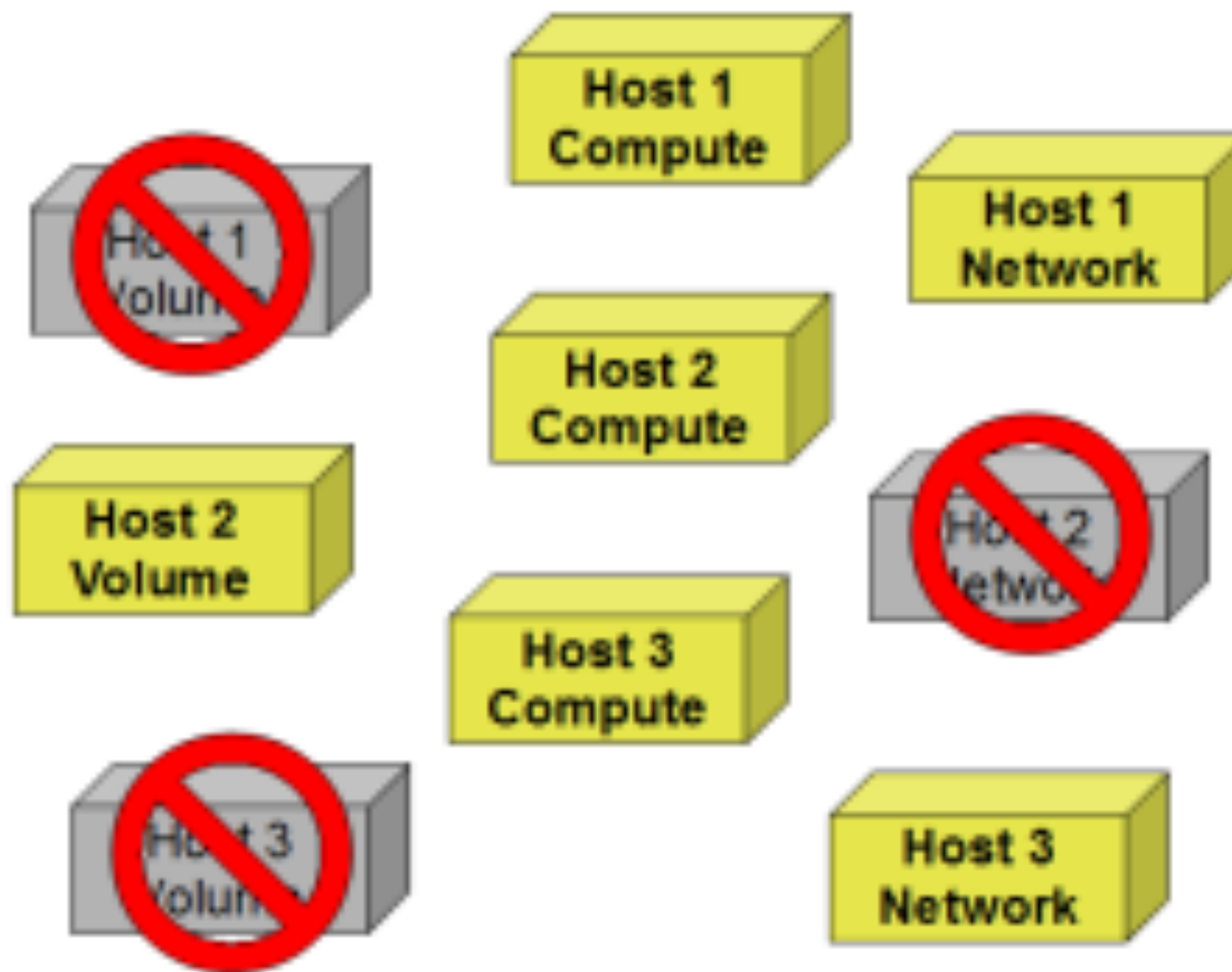
# Scheduler



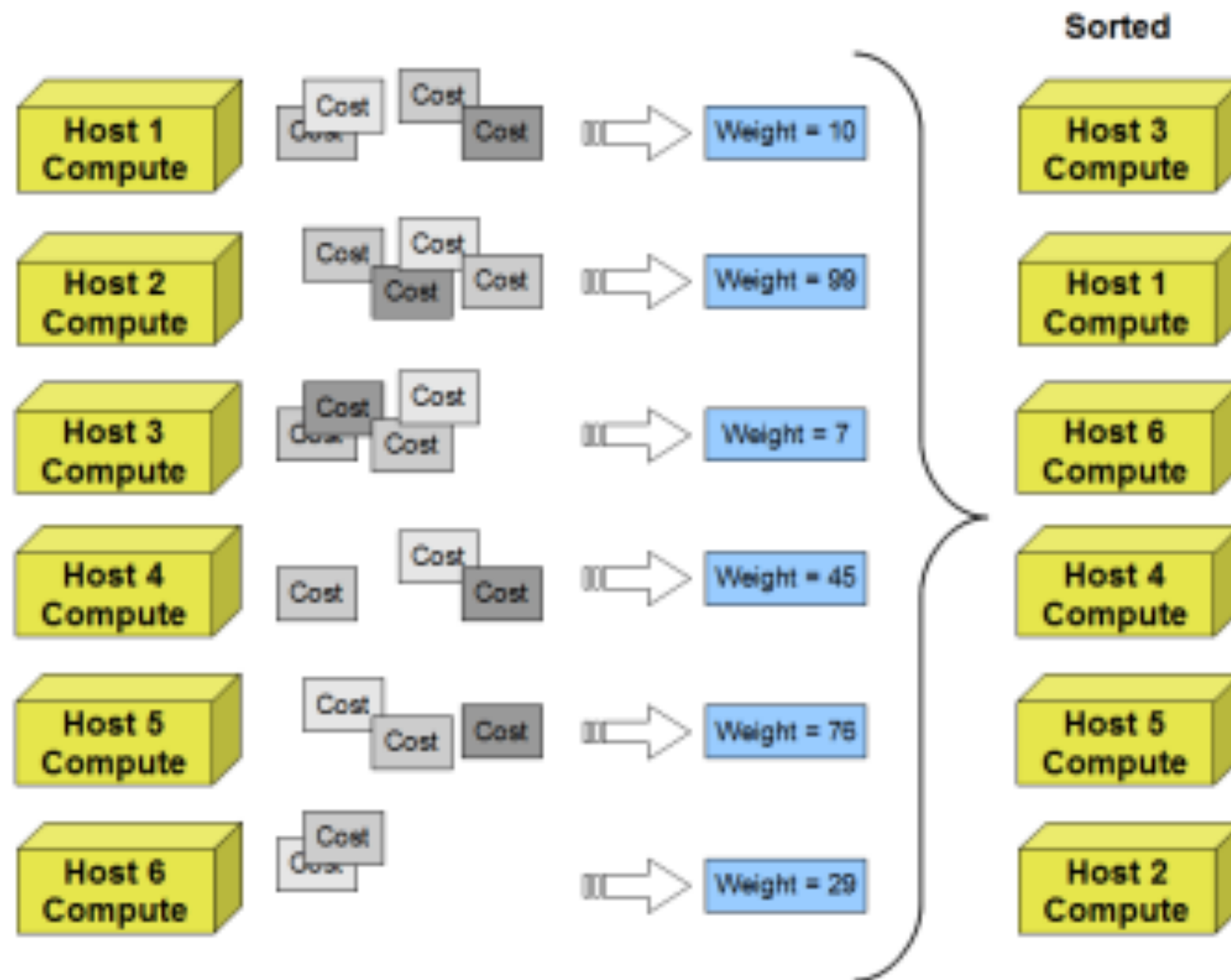
# Scheduler



# Filtering



# Cost & Weights





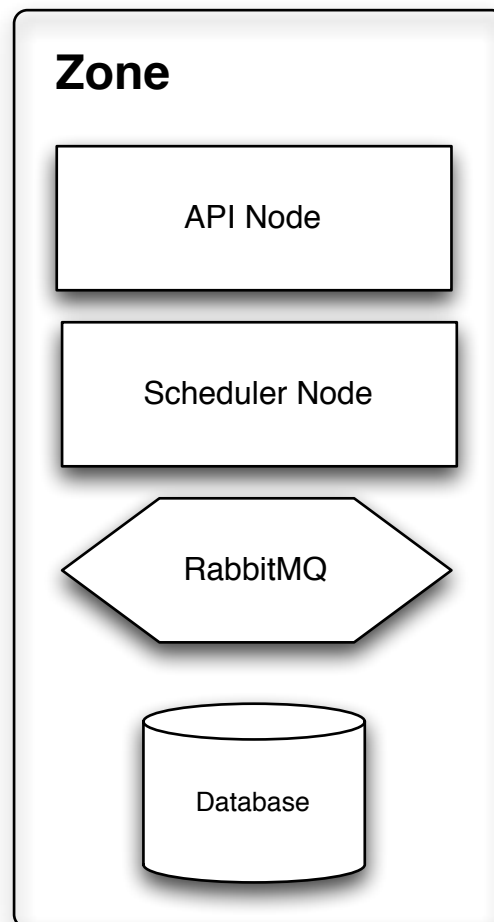
# Nova Cells

Where OpenStack is Headed

# Cells

- A Nova deployment is called a Cell
- A Cells allows you to partition your deployments into logical groups for load balancing and instance distribution

# Cells

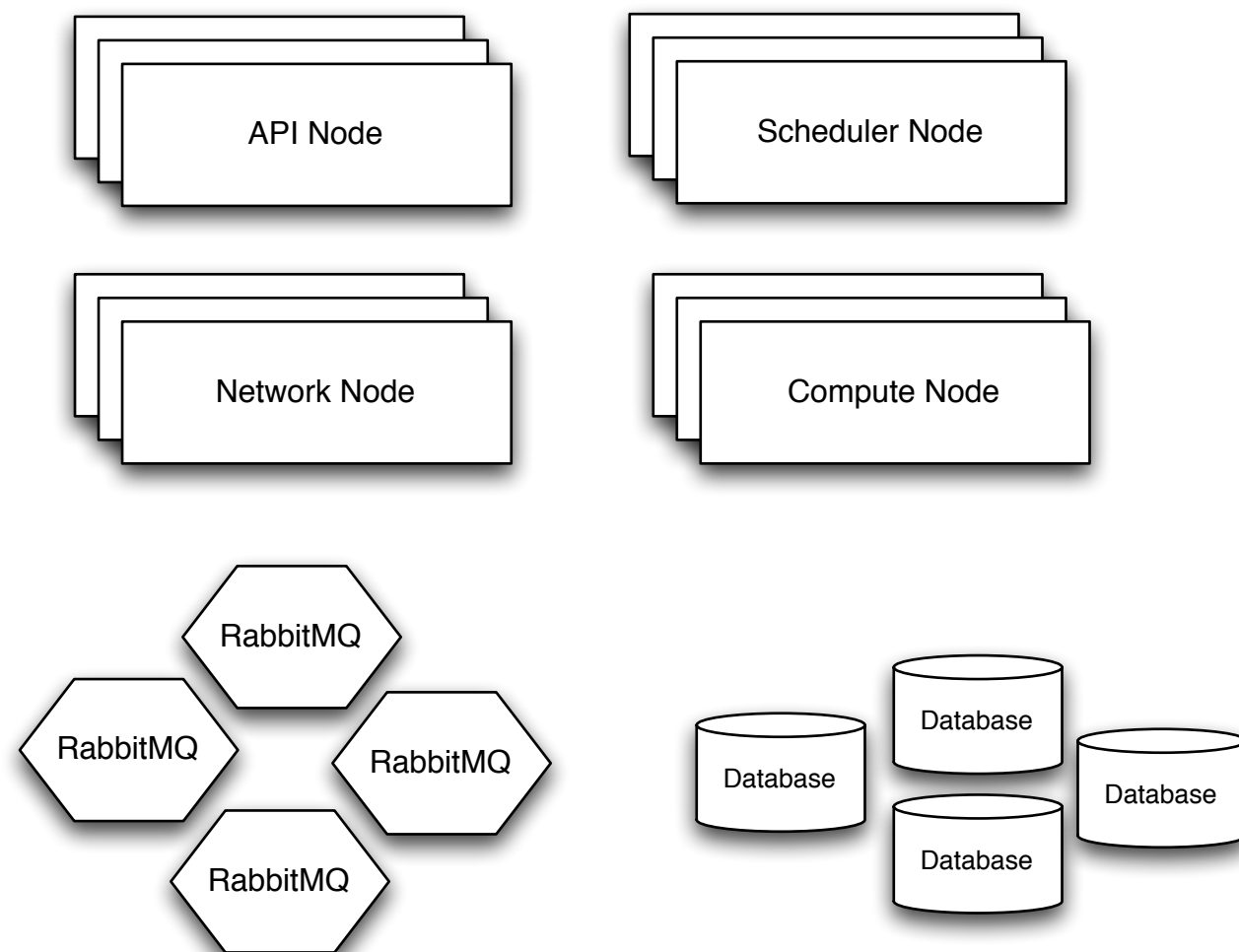


## Minimum Requirements for a Cell

- API Node
- Scheduler Node
- RabbitMQ
- Database

# Cells

## Zone

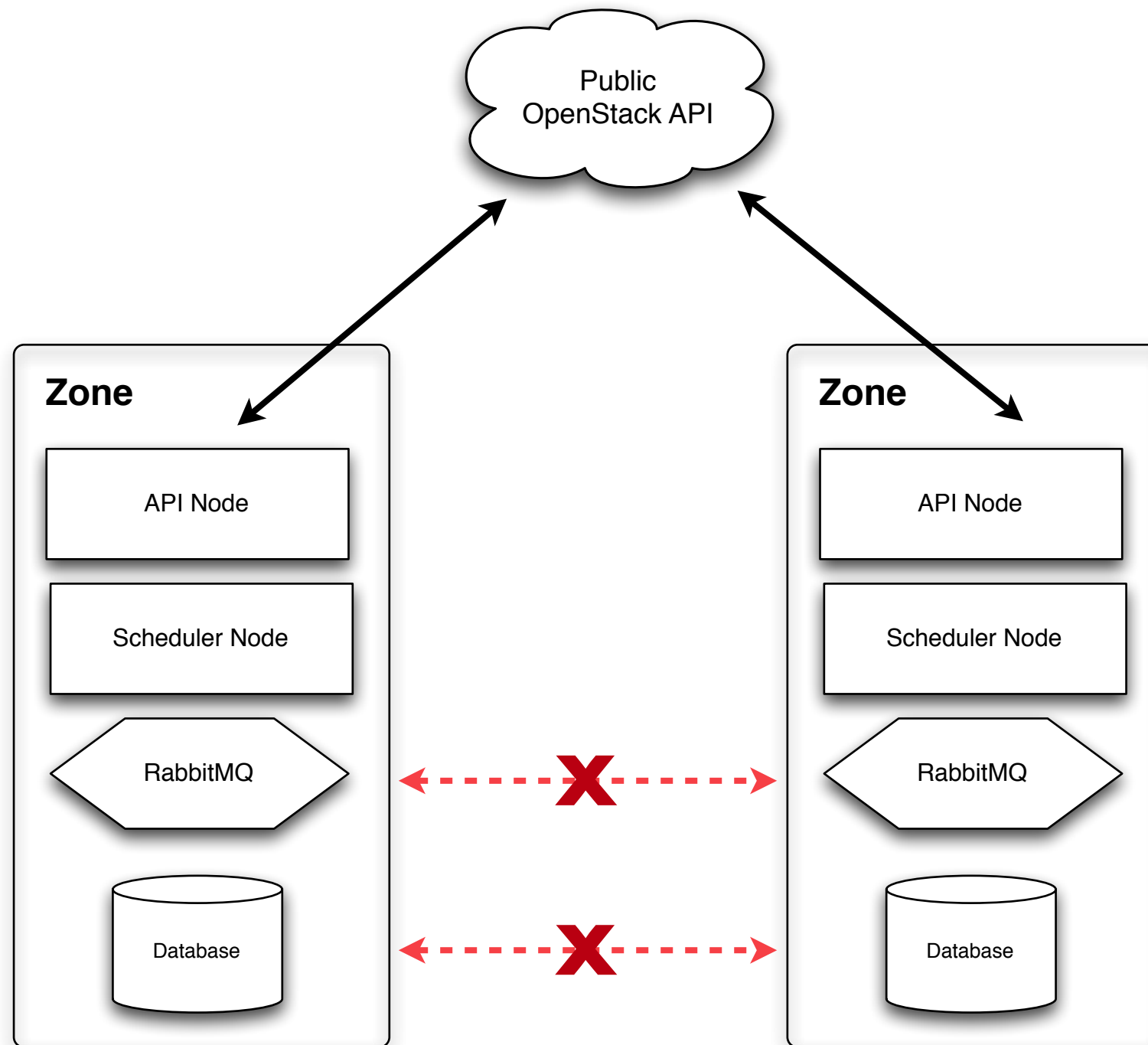


## A Complex Cell May Have

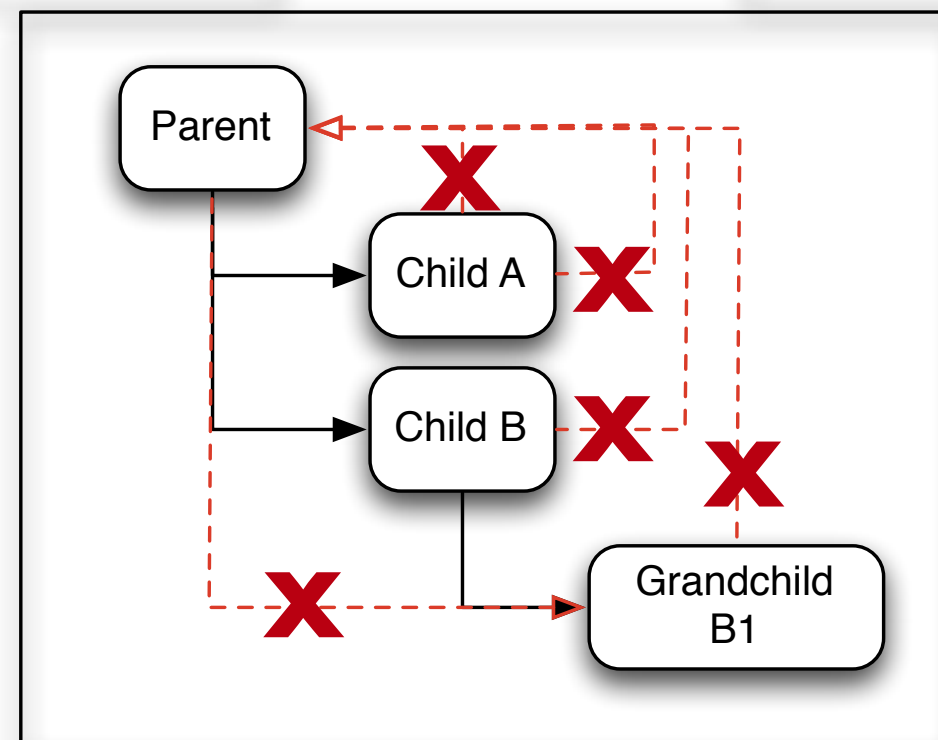
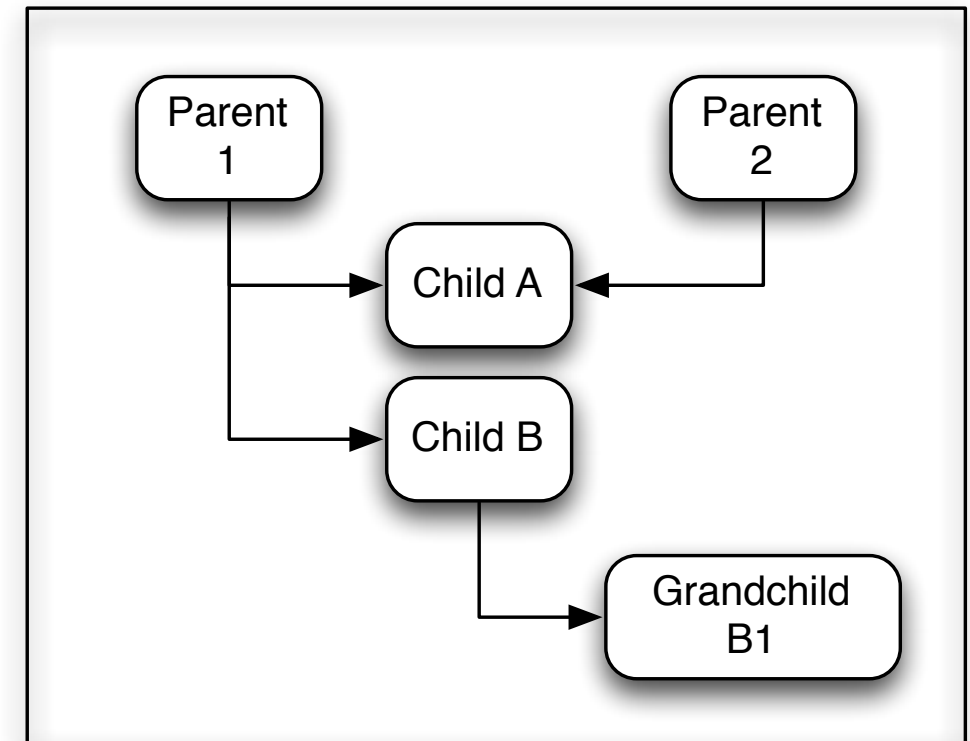
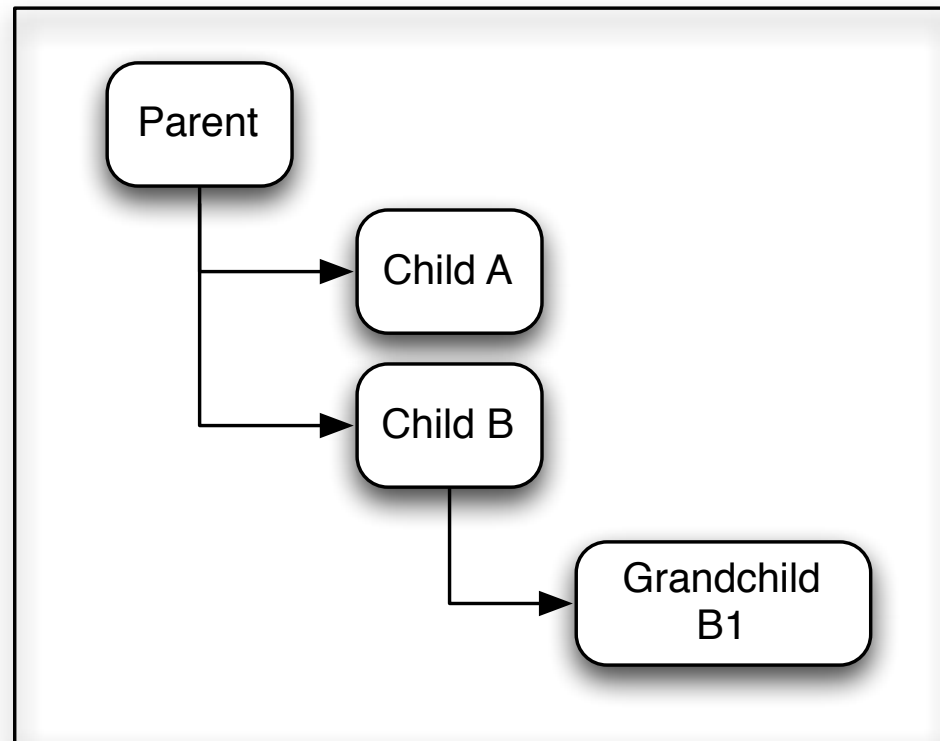
- Multiple API Nodes
- Multiple Scheduler Nodes
- Multiple Network Nodes
- Multiple Compute Nodes
- A RabbitMQ Cluster
- A Database Cluster

# Untrusted Cells

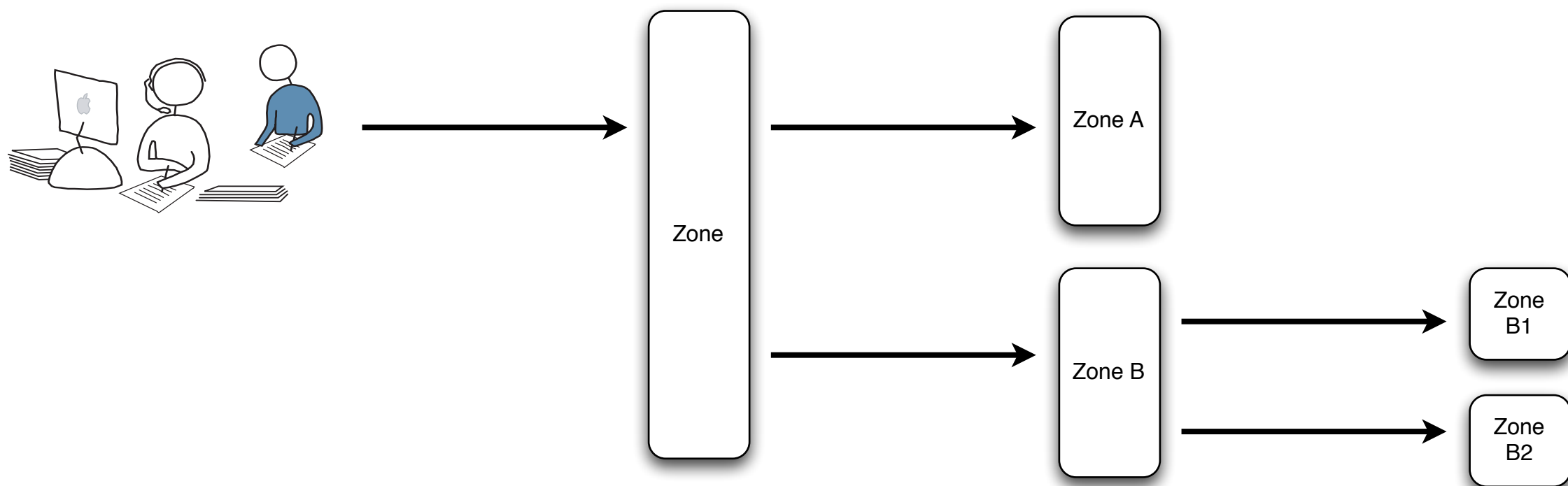
## Share Nothing



# Cell Nesting



# Request Processing

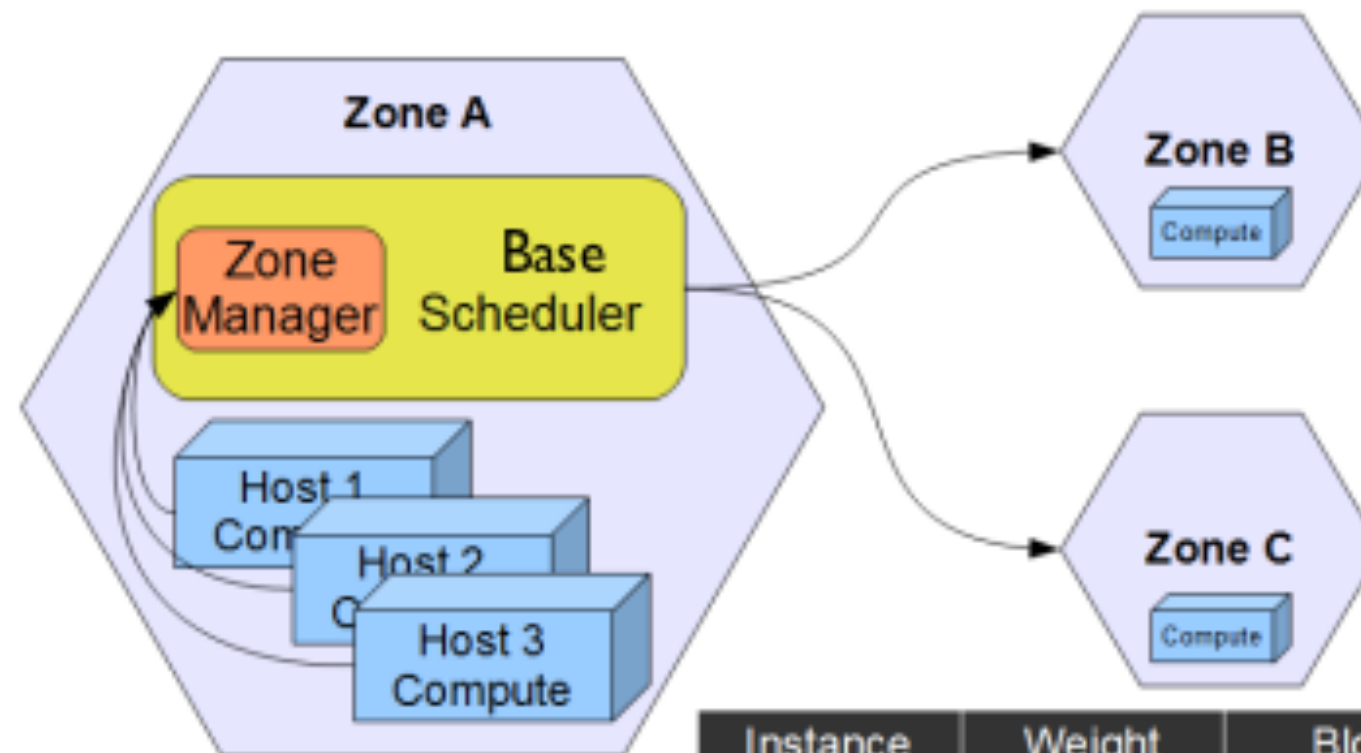


# Cost & Weights

`nova.compute.api.create_all_at_once()`

Quantity=2  
cpu > 20%  
disk > 50G  
net > 10mpbs  
gpu = True

Instance	Weight	Blob
1	23	f7b383c1...
2	78	cab6297...



Instance	Weight	Host
1	8	MyHost27
2	19	MyHost38

Instance	Weight	Blob
1	15	3ab34d7...
2	56	2746abd...



# Cell Capabilities

## General Capabilities

- `key=value;value;value, key=value;value;value`
- `hypervisor=xenserver;kvm,os=linux;windows`
- `--zone_capabilities` flag

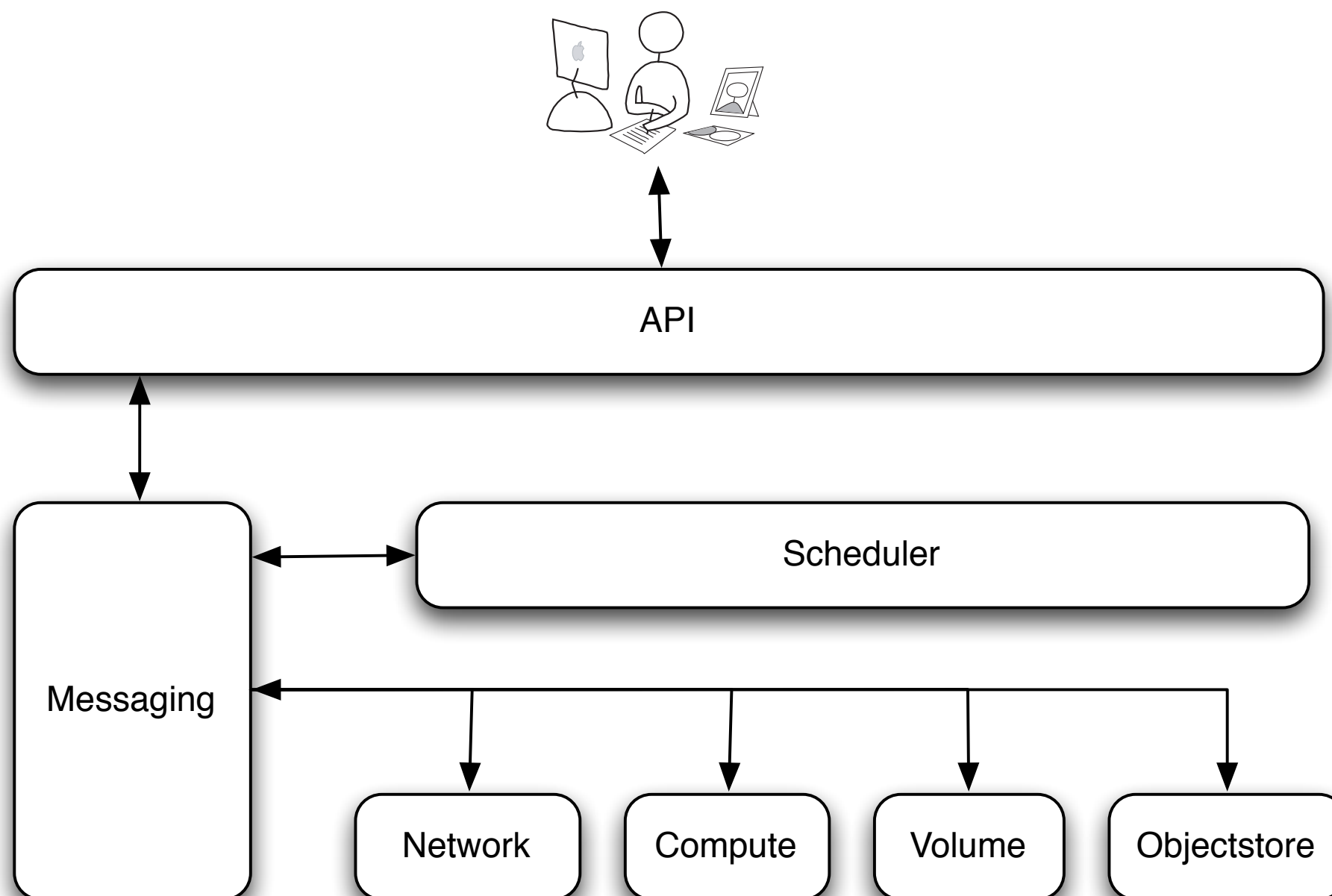
## Dynamic Capabilities

- Derived from *nova.manager.SchedulerDependentManager*
- set these capabilities by calling the *update\_service\_capabilities()* method on their *Manager* base class

# Nova Network Node

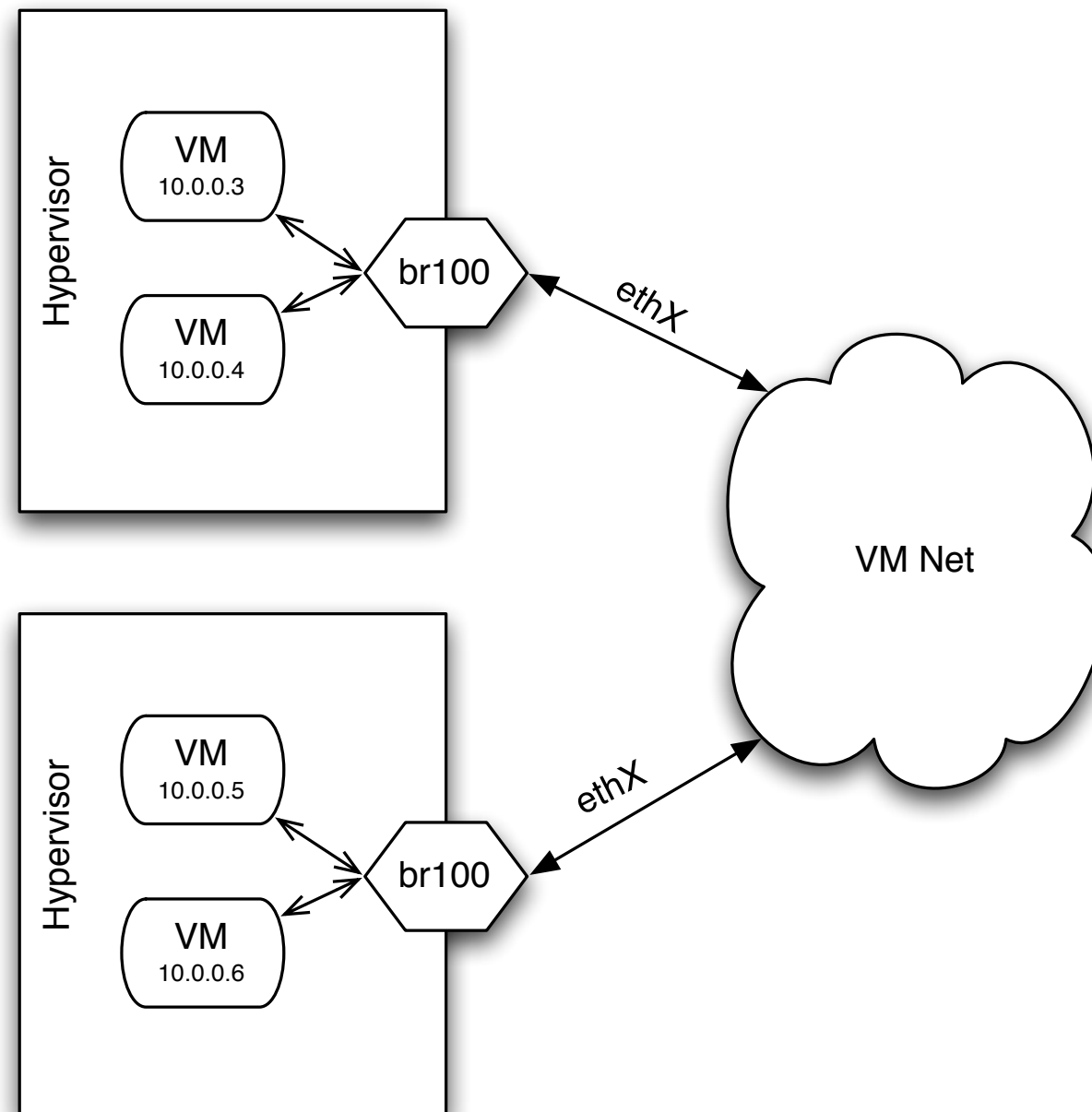
# Nova Network

## Request Flow



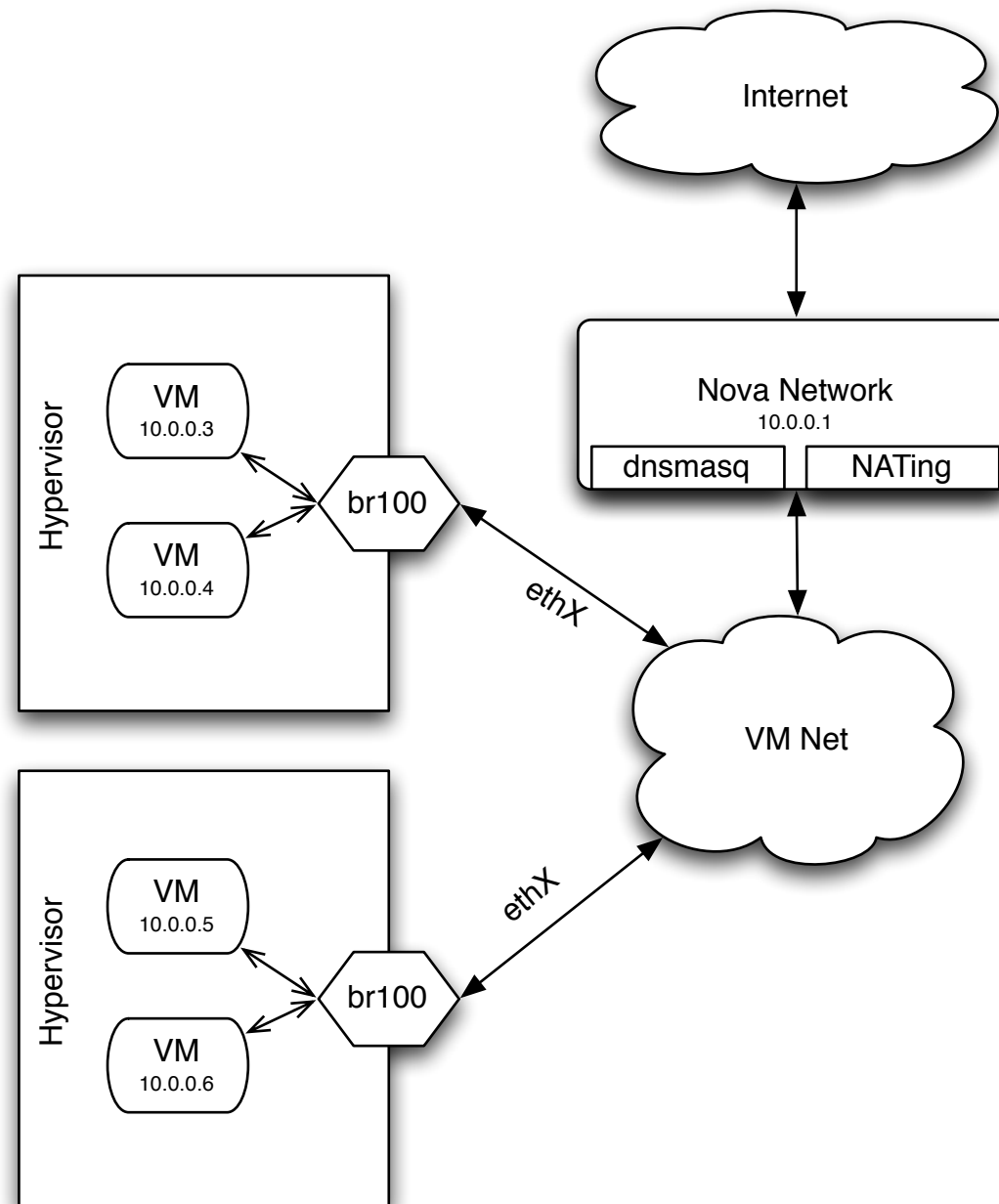
# Nova Network

## Flat Manager



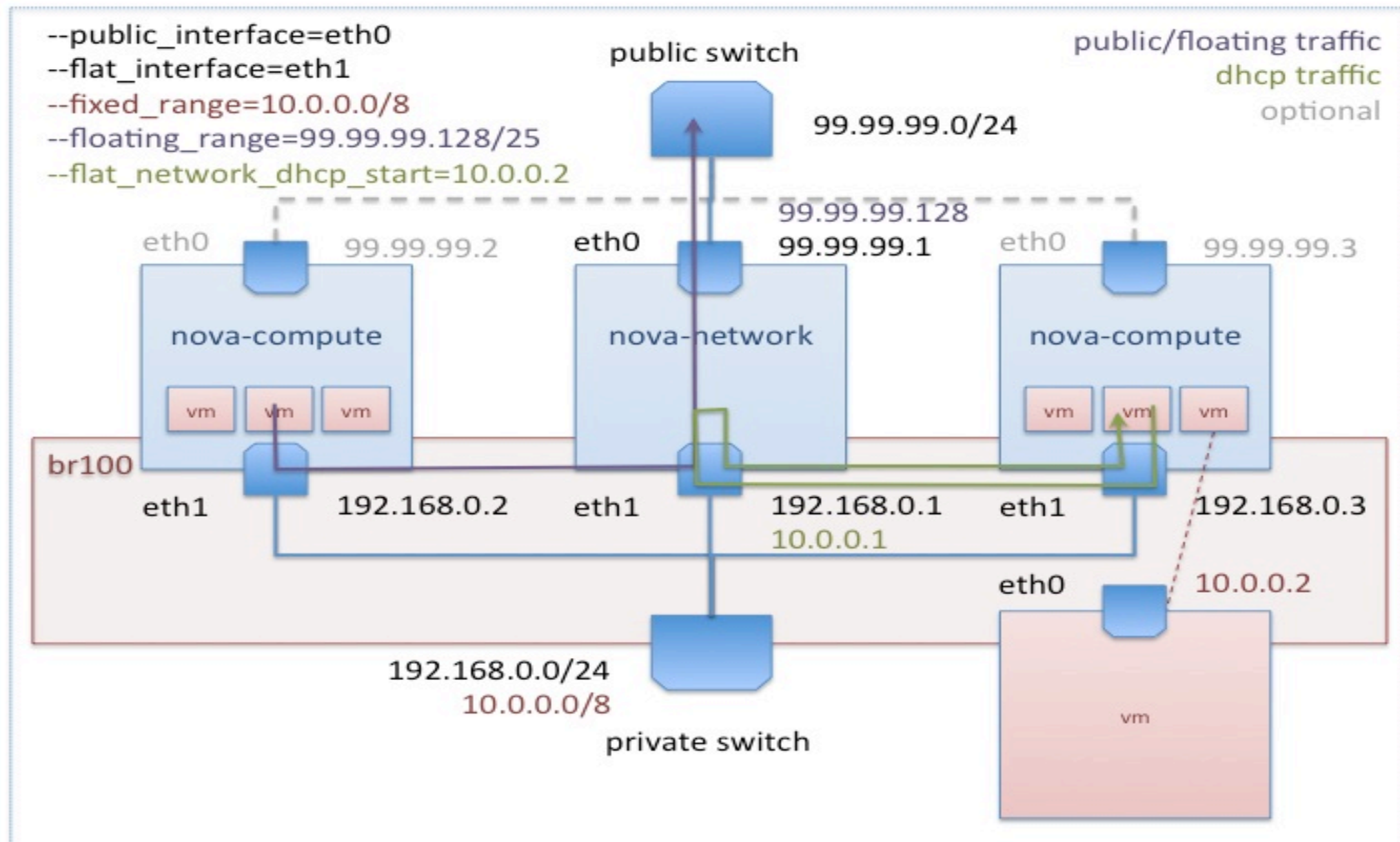
# Nova Network

## Flat DHCP Manager



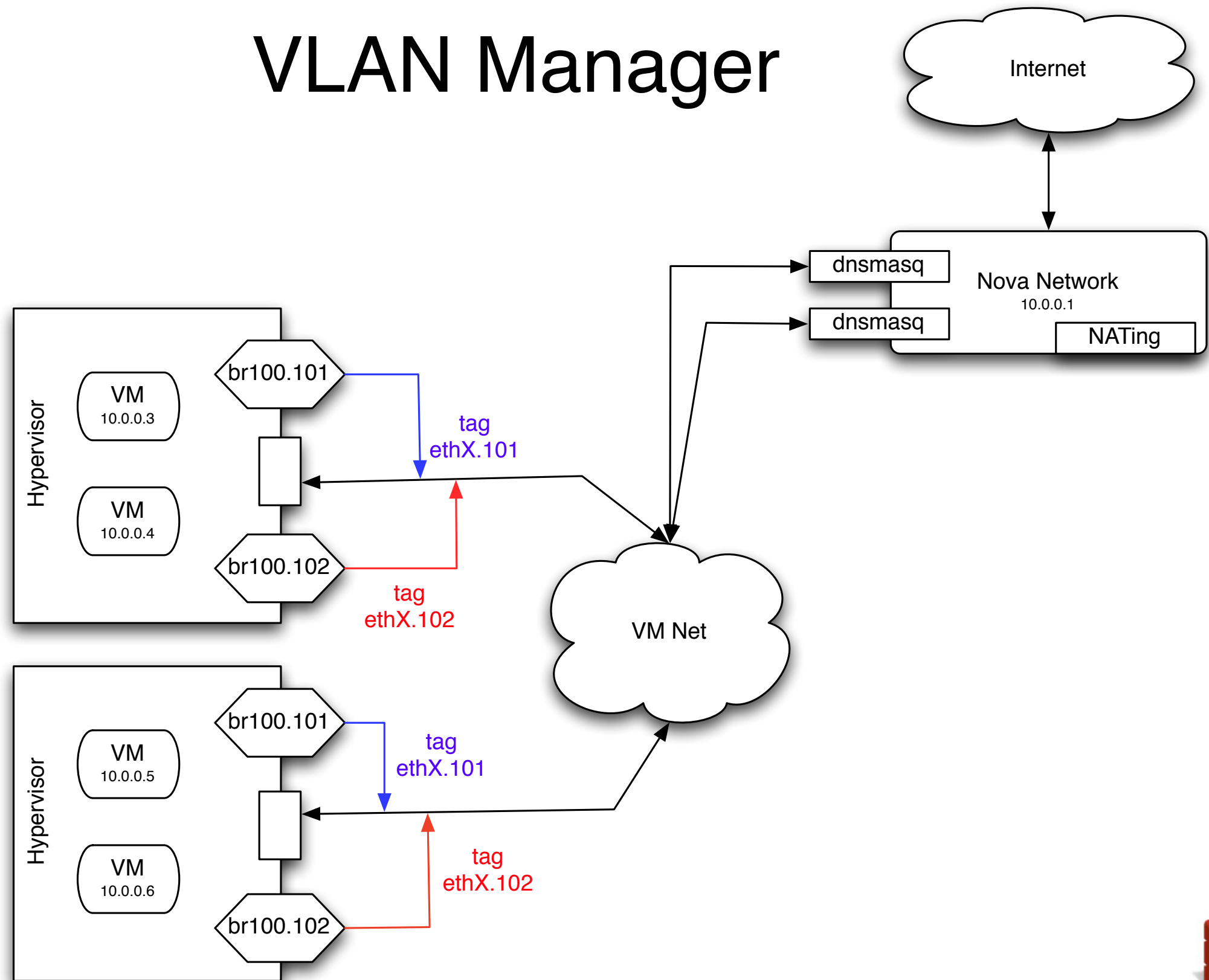
# Nova Network

## Flat DHCP Manager



# Nova Network

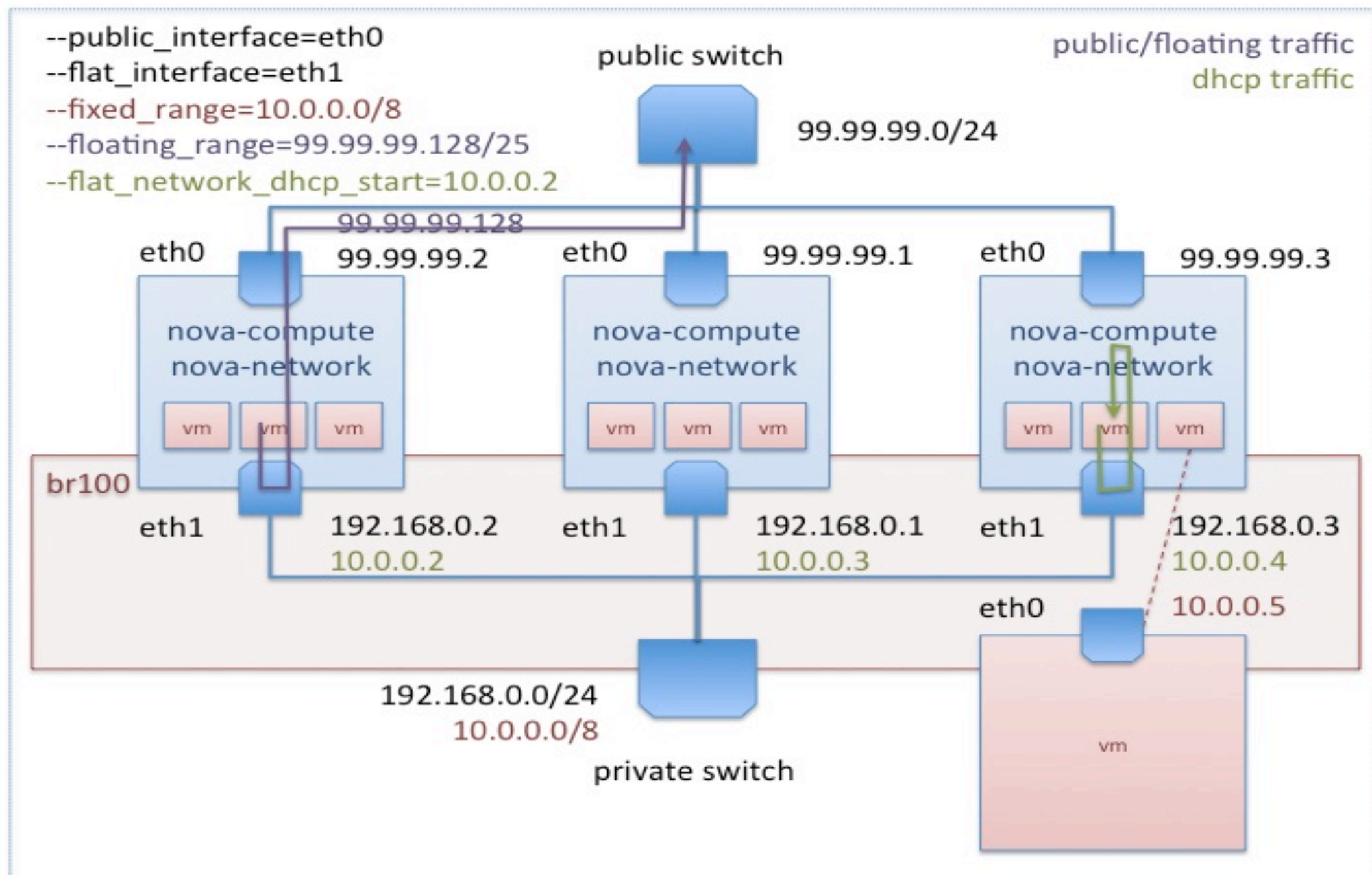
## VLAN Manager





# Nova Network

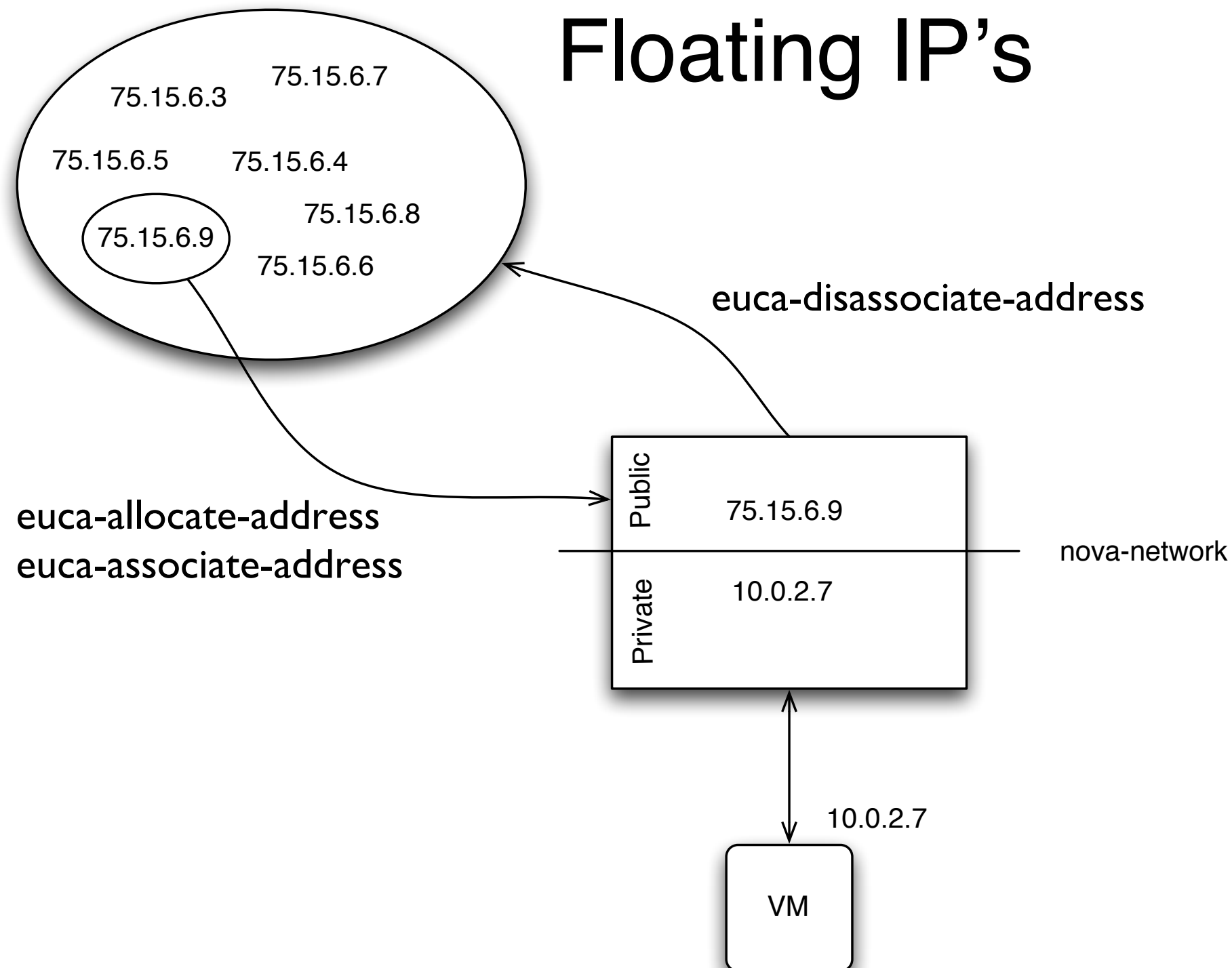
## High Availability Manager





# Nova Network

## Floating IP's



# Nova Network

## Floating IP's

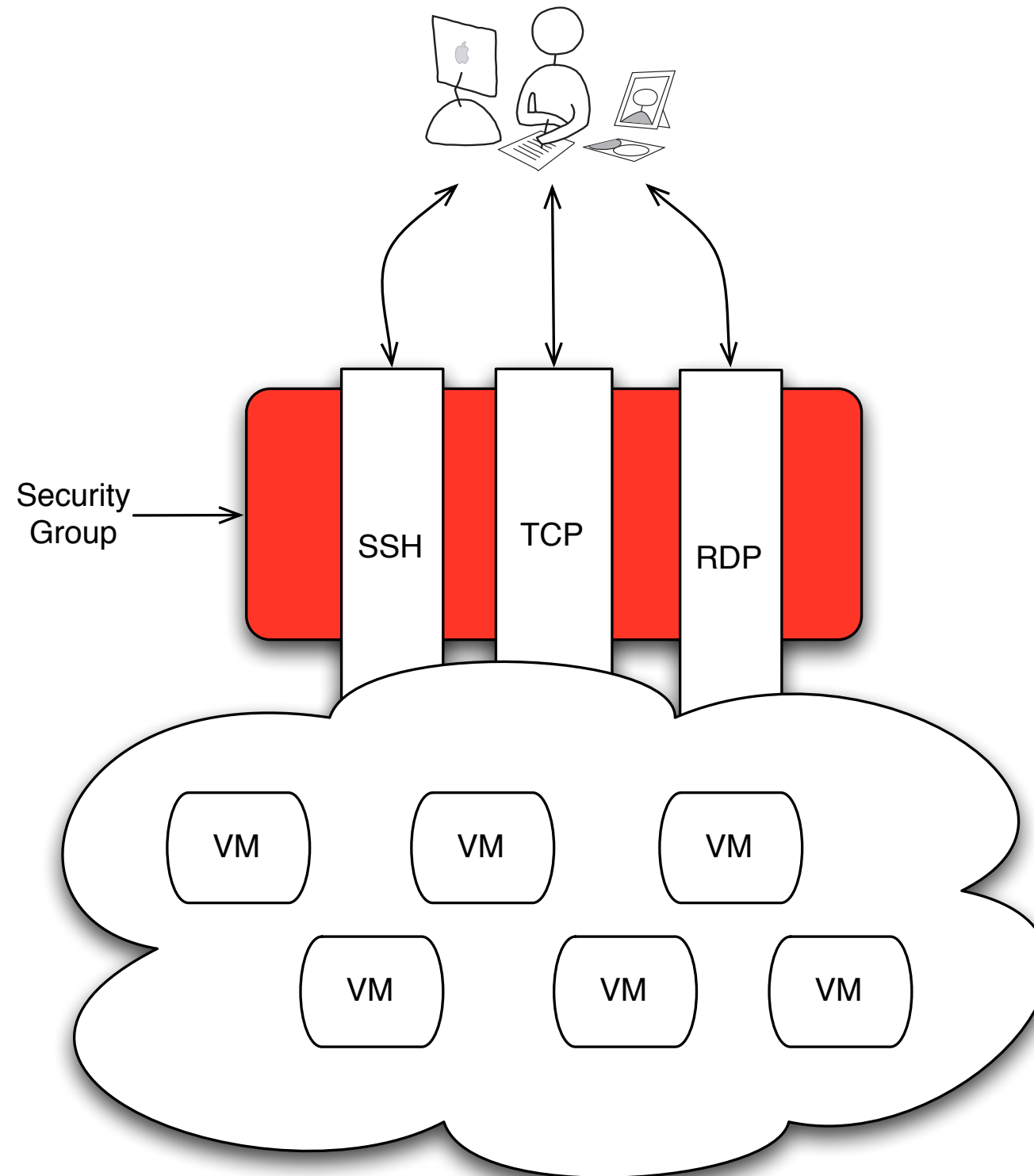
```
nova-manage floating create --ip_range=10.0.2.0/24
```

```
euca-allocate-address
```

```
euca-associate-address -i i-000000007 10.0.2.3
```

# Nova Network

## Security Groups



# Nova Network

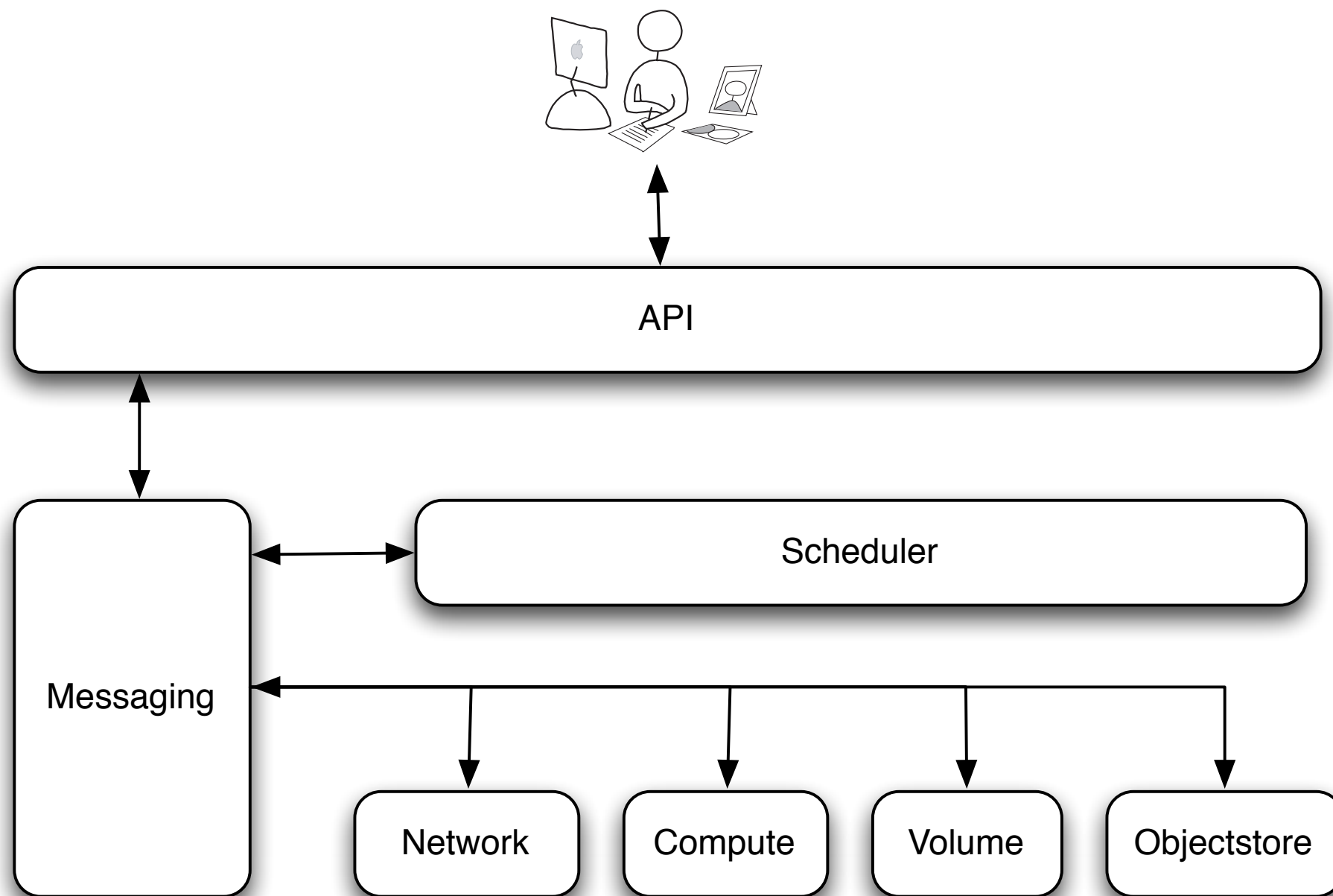
## Security Groups

- Create a new security group
  - `euca-add-group -d "my servers" myservers`
  - Add rules to the security group
    - ssh
      - `euca-authorize -P tcp -s 192.168.1.1 -p 22 myservers`
    - Ping
      - `euca-authorize -P icmp -s 192.168.1.1 -t -1:1 myservers`
- Boot an instance into the security group
  - `nova boot -flavor 3 -image 3 -ipgroup myservers <name>`

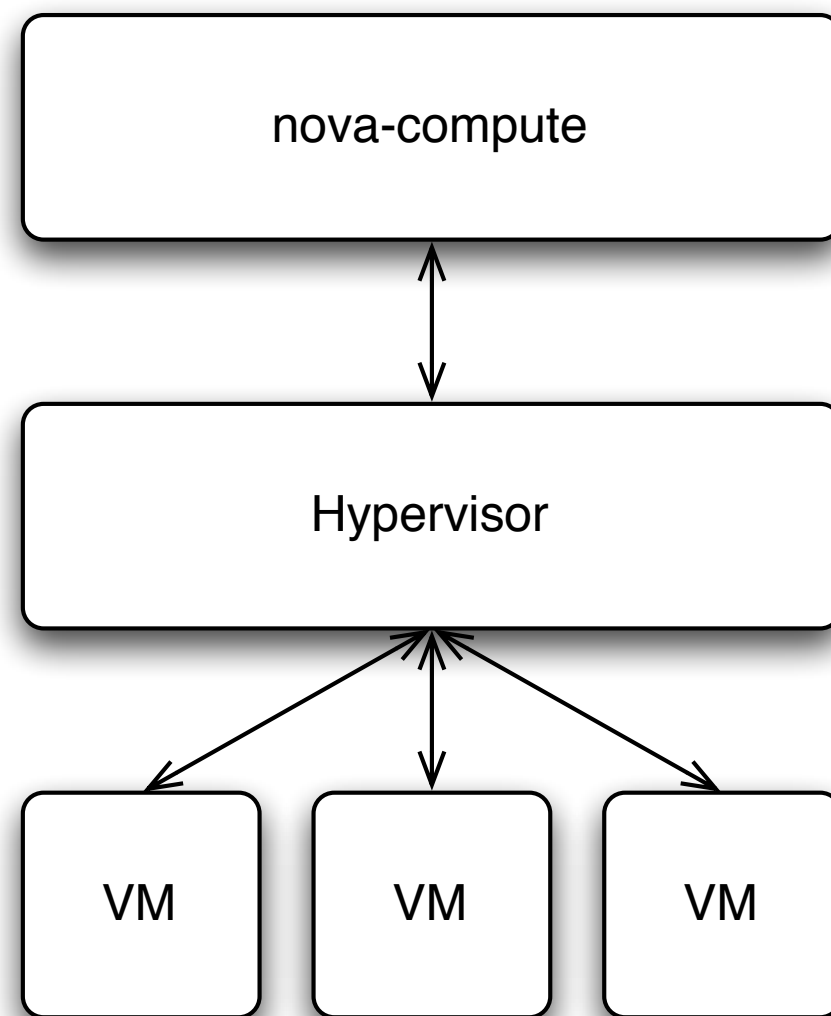
# Nova Compute Node

# Nova Compute Node

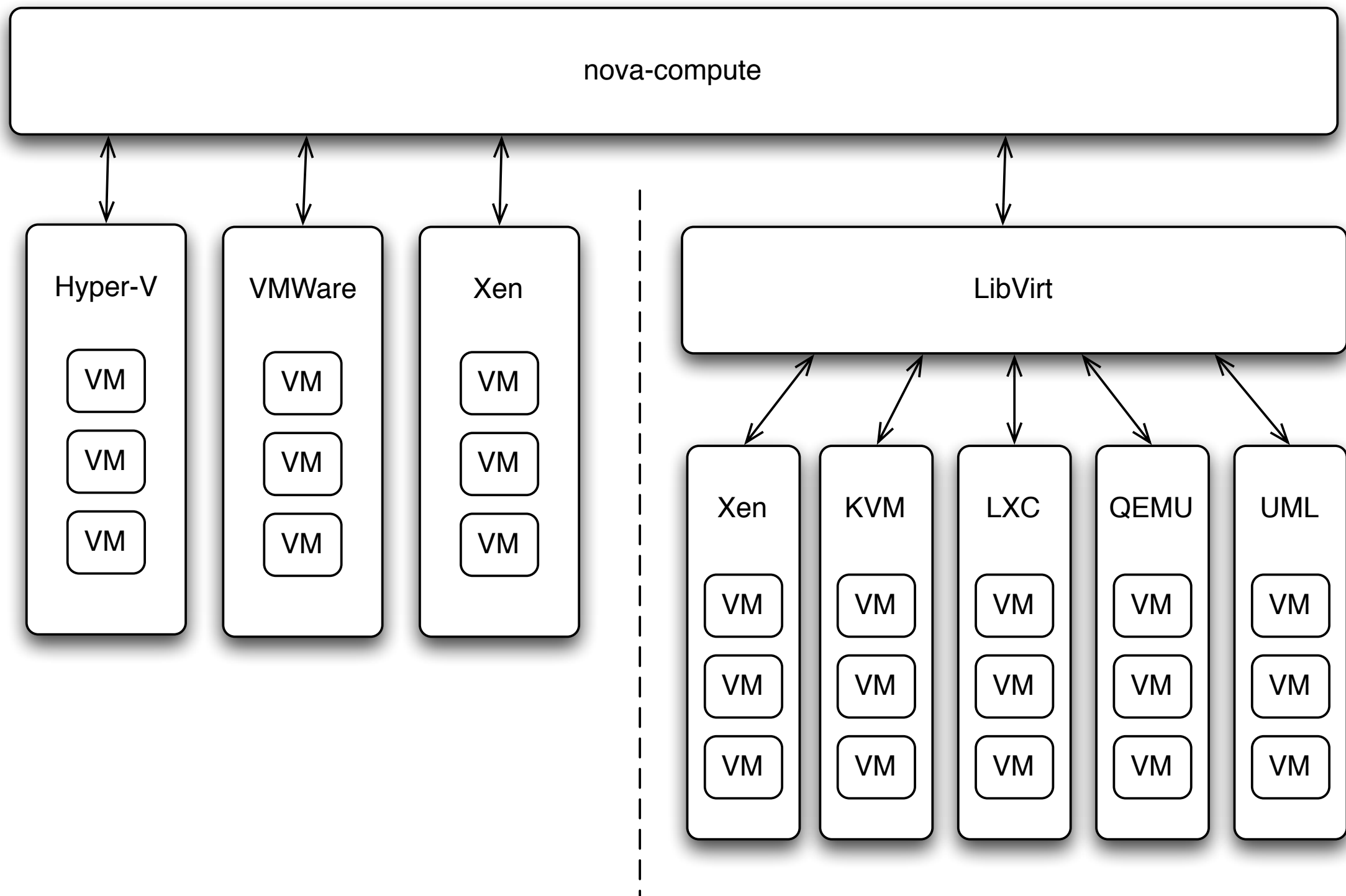
## Request Flow



# Nova Compute Node



# Nova Compute Node





# Nova Configuration

# Nova Configuration

- Flag File @ /etc/nova/nova.conf
- All Configuration options: **nova-manage config list**

# Exercise 0

Accessing the Classroom Cloud

# Accessing the Classroom Cloud

- ▶ Divide into Teams
- ▶ Training Lab WiFi Network
  - ▶ SSID: openstack
  - ▶ Password: openstack
- ▶ Lab Nodes
  - ▶ 192.168.2.<your team number> (example: 192.168.2.11)
  - ▶ username: openstack
  - ▶ password: openstack
- ▶ Shared nodes = one operator at a time
- ▶ Resist temptation
  - ▶ please don't hack the nodes belonging to other teams
  - ▶ try to stay with us and not jump ahead too far

# Accessing the Exercises

- ▶ Exercises will be available at:

<http://192.168.2.1/nova/>

# Exercise 1

## Nova All In One Installation

# The Nova Filesystem

# The Nova Filesystem

/etc/nova/

- api-paste.ini
- nova-compute.conf
- nova.conf

/var/log/nova/

- nova-api.log
- nova-compute.log
- nova-dhcpbridge.log
- nova-manage.log
- nova-network.log
- nova-objectstore.log
- nova-scheduler.log

/var/lib/nova/

- buckets
- CA
- extensions
- images
- instances
- keys
- networks
- nova.sqlite
- tmp



# The Nova Filesystem

```
/var/lib/nova/CA/
├── cacert.pem
├── certs
├── crl
├── crl.pem
├── index.txt
├── index.txt.attr
├── index.txt.old
├── INTER
├── newcerts
│   └── 10.pem
├── openssl.cnf
├── private
│   └── cakey.pem
├── reqs
├── serial
└── serial.old
```

```
/var/lib/nova/extensions/
├── admin -> /usr/share/pyshared/openstackx/admin
├── admin.py
├── api -> /usr/share/pyshared/openstackx/api
├── auth -> /usr/share/pyshared/openstackx/auth
├── compute -> /usr/share/pyshared/openstackx/compute
├── extras -> /usr/share/pyshared/openstackx/extras
└── __init__.py
```

# The Nova Filesystem

/var/lib/nova/instances/

- \_base
- instance-000000005
- instance-000000007

/var/lib/nova/instances/\_base

- 000000001
- 000000002
- 1b6453892473a467d07372d45eb05abc2031647a
- 77de68daecd823babbb58edb1c8e14d7106e83bb
- 77de68daecd823babbb58edb1c8e14d7106e83bb\_sm
- local\_40

/var/lib/nova/instances/instance-000000005/

- console.log
- disk
- disk.local
- kernel
- libvirt.xml
- ramdisk

/var/lib/nova/instances/instance-000000007/

- console.log
- disk
- disk.local
- libvirt.xml

# Nova

# Troubleshooting

# Nova Troubleshooting

- `/var/log/nova/nova-*.log`
- `ps -aux | grep nova`
- `virsh`
- `brctl`
- `rabbitmqctl`
- `sqlite3`, `mysql`, `psql`
- `euca-get-console-output`

# Nova CLI

# Nova CLI

## nova

actions	Retrieve server actions.
add-fixed-ip	Add new IP address to network.
boot	Boot a new server.
delete	Immediately shut down and delete a server.
diagnostics	Retrieve server diagnostics.
flavor-list	Print a list of available 'flavors' (sizes of servers).
image-create	Create a new image by taking a snapshot of a running server.
image-delete	Delete an image.
image-list	Print a list of available images to boot from.
list	List active servers.
migrate	Migrate a server.
pause	Pause a server.
reboot	Reboot a server.
rebuild	Shutdown, re-image, and re-boot a server.
remove-fixed-ip	Remove an IP address from a server.
rename	Rename a server.
rescue	Rescue a server.

# Nova CLI

## nova

resize	Resize a server.
resize-confirm	Confirm a previous resize.
resize-revert	Revert a previous resize (and return to the previous VM).
resume	Resume a server.
root-password	Change the root password for a server.
show	Show details about the given server.
suspend	Suspend a server.
unpause	Unpause a server.
unrescue	Unrescue a server.
zone	Show or edit a child zone. No zone arg for this zone.
zone-add	Add a new child zone.
zone-boot	Boot a new server, potentially across Zones.
zone-delete	Delete a zone.
zone-info	Get this zones name and capabilities.
zone-list	List the children of a zone.
help	Display help about this program or one of its subcommands.

# Nova CLI

## nova-manage

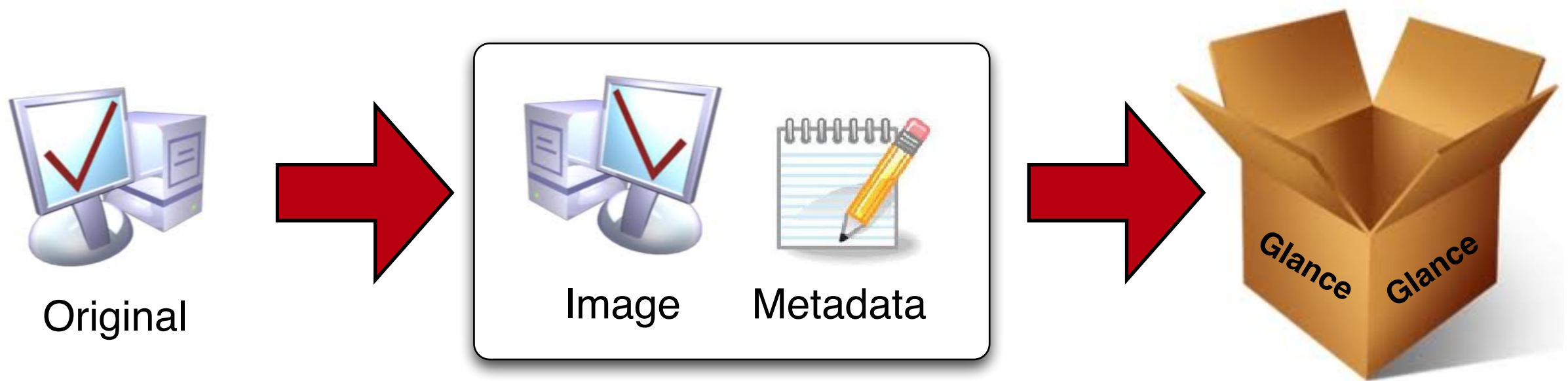
account	network
agent	project
config	role
db	service
drive	shell
fixed	user
flavor	version
floating	vm
host	volume
instance_type	vpn
image	vsa



# OpenStack Image Service

Codenamed: Glance

# Glance

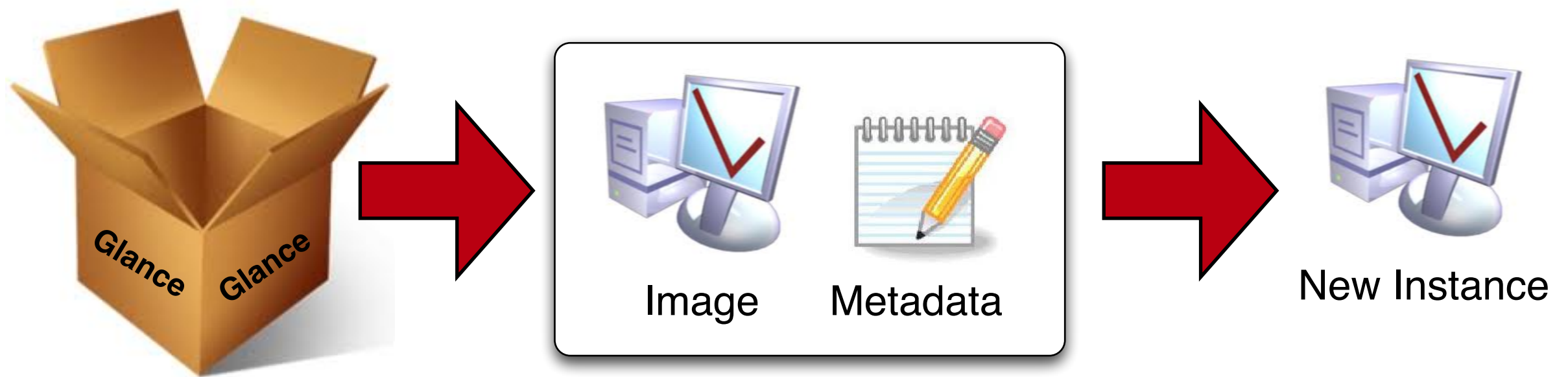


## im-age

n.

1. A reproduction of the form of a person or object, especially a sculptured likeness.
2. One that closely or exactly resembles another; a double:

# Glance

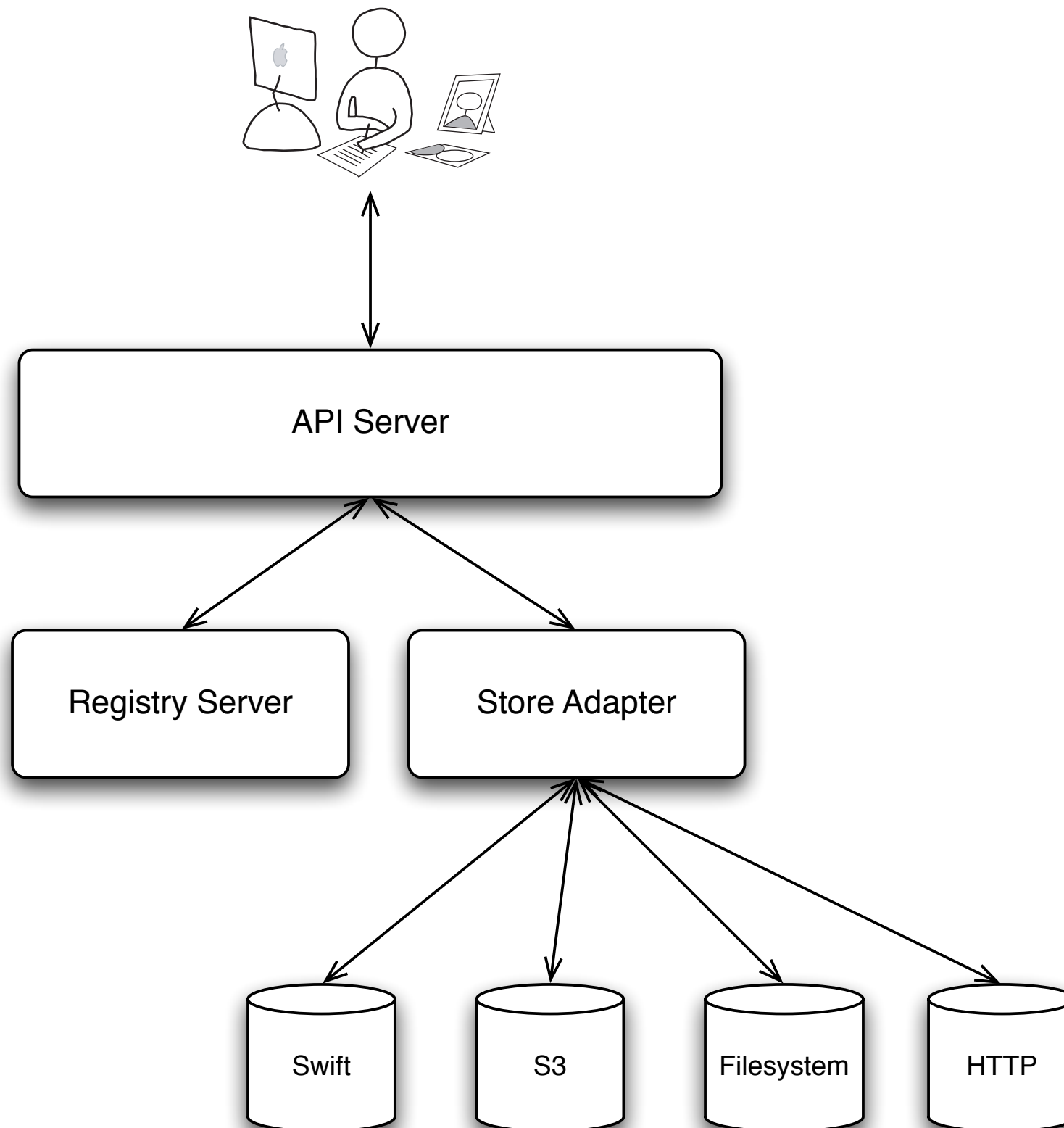


## im-age

n.

1. A reproduction of the form of a person or object, especially a sculptured likeness.
2. One that closely or exactly resembles another; a double:

# Glance



# Glance

## Image Identifiers

Images are uniquely identified by way of a URI that matches the following signature:

<Glance Server Location>/images/<ID>

The resource location of  
the Glance service

The image's identifier that is  
unique to that Glance server

# Glance

## Image Registry API

`http://<glance server location>:9292/v1`

GET	/images	Return brief information about public images
GET	/images/detail	Return detailed information about public images
HEAD	/images/<ID>	Return metadata about an image in HTTP headers
POST	/images	Register metadata about a new image
PUT	/images/<ID>	Update metadata about an existing image
DELETE	/images/<ID>	Remove an image's metadata from the registry

Image registries are any web service that adheres to the Glance REST-like API for image metadata.

# Glance

## Filtering Image Lists

<code>name=NAME</code>	Filters images having a <code>name</code> attribute matching <code>NAME</code> .
<code>container_format=FORMAT</code>	Filters images having a <code>container_format</code> attribute matching <code>FORMAT</code>
<code>disk_format=FORMAT</code>	Filters images having a <code>disk_format</code> attribute matching <code>FORMAT</code>
<code>status=STATUS</code>	Filters images having a <code>status</code> attribute matching <code>STATUS</code>
<code>size_min=BYTES</code>	Filters images having a <code>size</code> attribute greater than or equal to <code>BYTES</code>
<code>size_max=BYTES</code>	Filters images having a <code>size</code> attribute less than or equal to <code>BYTES</code>
<code>sort_key=KEY</code>	Results will be ordered by the specified image attribute <code>KEY</code> . Accepted values include <code>id</code> , <code>name</code> , <code>status</code> , <code>disk_format</code> , <code>container_format</code> , <code>size</code> , <code>created_at</code> (default) and <code>updated_at</code> .
<code>sort_dir=DIR</code>	Results will be sorted in the direction <code>DIR</code> . Accepted values are <code>asc</code> for ascending or <code>desc</code> (default) for descending.

# Glance

## POST /images

```
{ 'image':  
  { 'id': <ID> | None,  
    'name': <NAME>,  
    'status': <STATUS>,  
    'disk_format': <DISK_FORMAT>,  
    'container_format': <CONTAINER_FORMAT>,  
    'properties': [ ... ]  
  }  
}
```

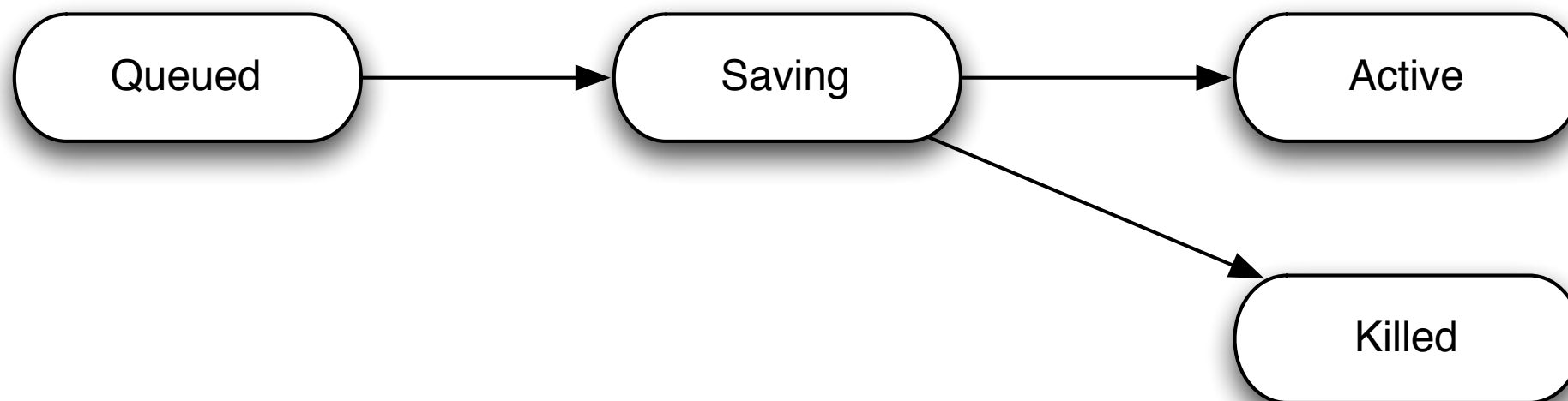
- status must be non-empty, and must be one of **active**, **saving**, **queued**, or **killed**
- disk\_format must be non-empty, and must be one of **ari**, **aki**, **ami**, **raw**, **iso**, **vhd**, **vdi**, **qcow2**, or **vmdk**
- container\_format must be non-empty, and must be one of **ari**, **aki**, **ami**, **bare**, or **ovf**
- If disk\_format *or* container\_format is **ari**, **aki**, **ami**, then *both* disk\_format and container\_format must be the same.



# Glance

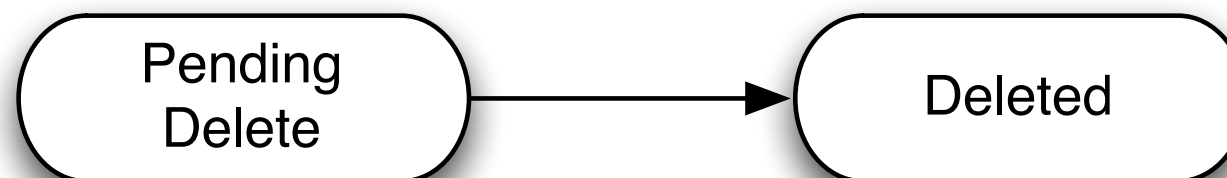
## Image Statuses

### Adding Images



---

### Removing Images



# Glance

## Disk Formats

<b>raw</b>	This is an unstructured disk image format
<b>vhd</b>	This is the VHD disk format, a common disk format used by virtual machine monitors from VMWare, Xen, Microsoft, VirtualBox, and others
<b>vmdk</b>	Another common disk format supported by many common virtual machine monitors
<b>vdi</b>	A disk format supported by VirtualBox virtual machine monitor and the QEMU emulator
<b>iso</b>	An archive format for the data contents of an optical disc (e.g. CDROM)
<b>qcow2</b>	A disk format supported by the QEMU emulator that can expand dynamically and supports Copy on Write
<b>aki</b>	This indicates what is stored in Glance is an Amazon kernel image
<b>ari</b>	This indicates what is stored in Glance is an Amazon ramdisk image
<b>ami</b>	This indicates what is stored in Glance is an Amazon machine image

# Exercise 2

## Glance Installation & Image Loading

# Exercise 3

## Boot Instance

# OpenStack Identity Service

Codenamed: Keystone

# Keystone Concepts



Role



User



Credentials



Authentication



Endpoint



Service



Tenant



Token

# Keystone Concepts

## User



Person



System



Service

# Keystone Concepts

## Credentials





# Keystone Concepts

## Authentication



# Keystone Concepts

## Token



# Keystone Concepts

## Tenant



# Keystone Concepts

## Service



# Keystone Concepts

## Endpoint



# Keystone Concepts

## Role



# Keystone

## Populating Auth Data

### Add Tenants

- `keystone tenant-create --name MyTenant`
- `keystone tenant-list`

### Add Users

- `keystone user-create --tenant_id <GUID> --name myuser --pass mypassword`
- `keystone user-list`

### Add Roles

- `keystone role-create --name Admin`
- `keystone role-create --name Member`
- `keystone role-list`

### Grant Roles

- `keystone user-role-add --user <GUID> --tenant_id <GUID> --role <GUID>`

# Keystone

## Populating Auth Admin Data

### Keystone Service Admin

- `keystone service-create --name=keystone --type=identity --description="Identity Service"`

### Endpoint Templates

- `keystone endpoint-create --region RegionOne --service_id=<GUID> --publicurl=http://127.0.0.1:5000/v2.0 --internalurl=http://127.0.0.1:5000/v2.0 --adminurl=http://127.0.0.1:35357/v2.0`

### Add Tokens

- `keystone-manage token add 999888777666 myadmin MyTenant 2015-02-05T00:00`

### Add Credentials

- `keystone-manage credentials add admin EC2 admin:admin mypassword MyTenant`



# Exercise 4

## Keystone Installation & Setup

# OpenStack Dashboard

The OpenStack Web Interface

# Exercise 5

## Dashboard Installation



# Demo

## A Dashboard Walkthrough

# Packages for Labs

deb <http://ops.rcb.me/packages> maverick diablo-d5

# Resources

- <http://www.openstack.org>
- <https://launchpad.net/openstack>
- <https://github.com/openstack>
- <https://github.com/cloudbuilders>
- <http://www.referencearchitecture.org/>
- <http://devstack.org/>
- <http://programmerthoughts.com/>
- <http://www.unchainyourbrain.com>
- <http://www.tlohg.com>

# Tony Campbell

- [tony.campbell@rackspace.com](mailto:tony.campbell@rackspace.com)
- @tonytcampbell (twitter)
- +Tony Campbell (google+)
- 1 210-312-4150

# John McKenzie

- [john.mckenzie@rackspace.com](mailto:john.mckenzie@rackspace.com)
- @jmckind (twitter)
- +John McKenzie (google+)
- 1 210-312-4667

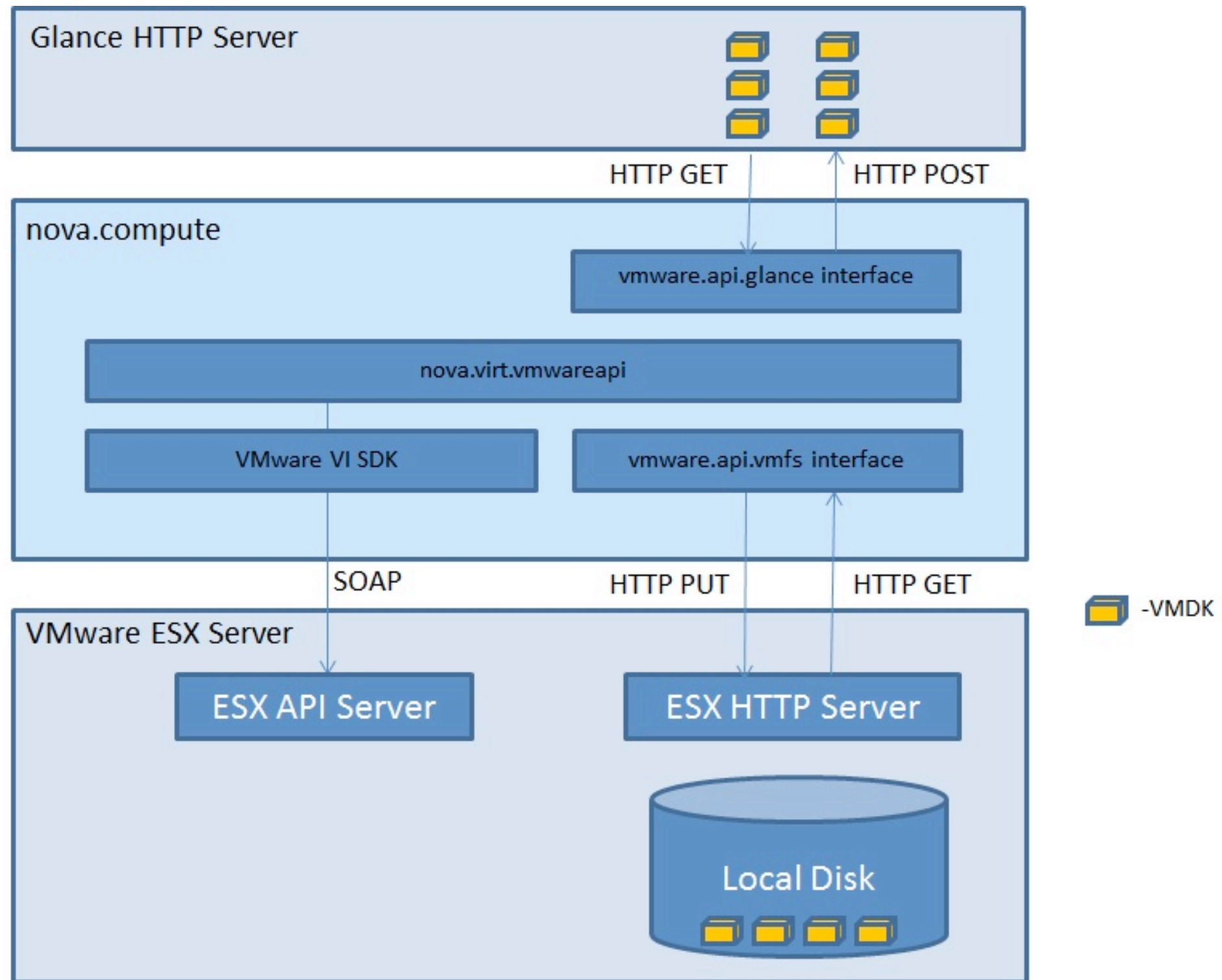


# Byron McCollum

- [byron.mccollum@rackspace.com](mailto:byron.mccollum@rackspace.com)
- @bgmccollum (twitter)
- 1 210-312-4204

# Appendix

# Nova & VMWare



[http://nova.openstack.org/vmwareapi\\_readme.html](http://nova.openstack.org/vmwareapi_readme.html)